Multiresolution Neural Networks for Image Edge Detection and Restoration

Horng-Chang Yang B. Sc. M. Sc.

A thesis submitted to
The University of Warwick
for the degree of
Doctor of Philosophy

Comp. Sci.

August 1994

Multiresolution Neural Networks for Image Edge Detection and Restoration

Horng-Chang Yang

A thesis submitted to
The University of Warwick
for the degree of
Doctor of Philosophy

August 1994

Summary

One of the methods for building an automatic visual system is to borrow the properties of the human visual system (HVS). Artificial neural networks are based on this doctrine and they have been applied to image processing and computer vision. This work focused on the plausibility of using a class of Hopfield neural networks for edge detection and image restoration.

To this end, a quadratic energy minimization framework is presented. Central to this framework are relaxation operations, which can be implemented using the class of Hopfield neural networks. The role of the uncertainty principle in vision is described, which imposes a limit on the simultaneous localisation in both class and position space. It is shown how a multiresolution approach allows the trade-off between position and class resolution and ensures both robustness in noise and efficiency of computation. As edge detection and image restoration are ill-posed, some *a priori* knowledge is needed to regularize these problems. A multiresolution network is proposed to tackle the uncertainty problem and the regularization of these ill-posed image processing problems. For edge detection, orientation information is used to construct a compatibility function for the strength of the links of the proposed Hopfield neural network.

Edge detection results are presented for a number of synthetic and natural images which show that the iterative network gives robust results at low signal-to-noise ratios (0 dB) and is at least as good as many previous methods at capturing complex region shapes. For restoration, mean square error is used as the quadratic energy function of the Hopfield neural network. The results of the edge detection are used for adaptive restoration. Also shown are the results of restoration using the proposed iterative network framework.

Key Words:

Multiresolution,Hopfield neural network,Edge detection,Orientation, Restoration

Contents

List of Figures

List of Tables

Acknowledgements

This work was conducted within the Image and Signal Processing Research Group in the Department of Computer Science at Warwick University. I am grateful to the Ministry of Education of the Republic of China on Taiwan for its financial support.

I would like to thank all the staff at the Computer Science Department. In particular, thanks go to all my friends and colleagues, past and present, of the Image and Signal Processing Group at Warwick: Abhir Bhalerao, Andrew Calway, Simon Clippingdale, Nicola Cross, Andrew Davies, Wooi Boon Goh, Tao-I Hsu, Andrew King, Peter Meulemans, Edward Pearson, Hugh Scott, Tim Shuttleworth, Martin Todd and June, Wong. They have made numerous contributions and have provided a stimulating and friendly environment in which to work. Thanks also to Jeff Smith for providing essential software support. Moreover, they have made my stay in England a precious and memorable experience.

I am particularly indebted to my supervisor Dr. Roland Wilson, without whose ideas, enthusiasm, profound expertise in the subject and patience in correcting my English, this work would not have been possible.

Declaration

I declare that, except where specifically acknowledged, the material contained in this thesis is my own work and that it has neither been previously published nor submitted elsewhere for the purpose of obtaining an academic degree.

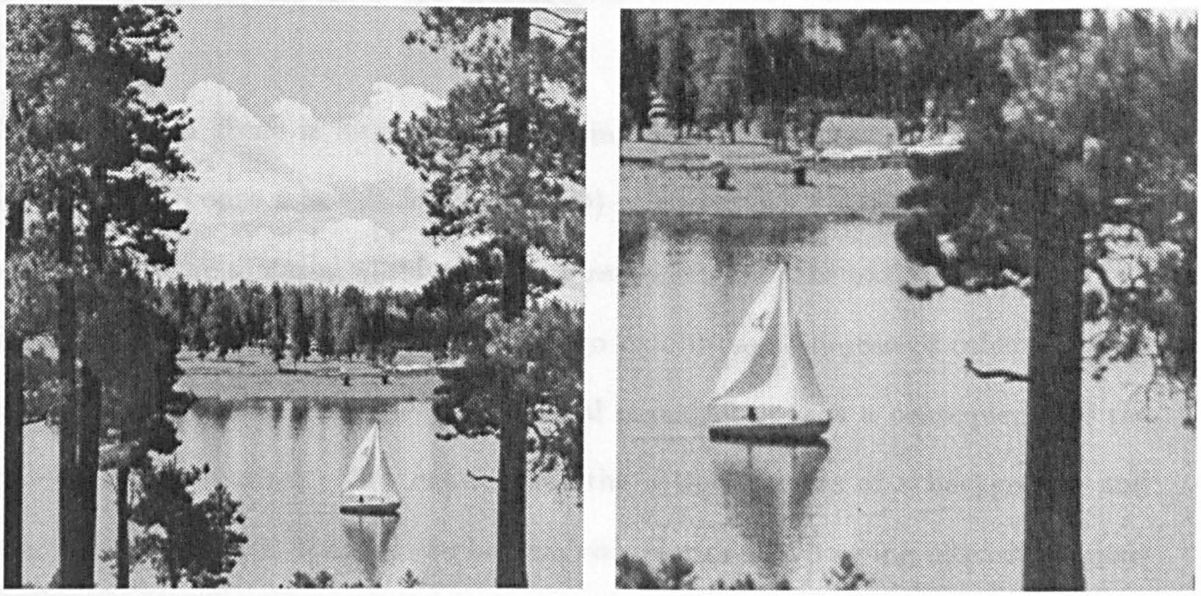
Horng-Chang Yang

Chapter 1

Introduction

1.1 Introductory Remarks

In the past three decades, many researchers have been striving for a computer vision system which can automate many tasks involving seeing. The task is to use image capturing equipment in place of human eyes, and a computer and algorithms in place of the little understood human brain. While computer vision systems may have a degree of success in a controlled environment in the recognition of simple parts such as bolts and cogs, in the general case, the system is far from the performance of the human visual system (HVS). Seeing seems to us so natural that its complexity is easily underestimated. Take a look at figure 1.1. A human being has no difficulty in recognising this picture as lake scenery, even if he or she has not seen the image before. While using a digital computer, to the author's knowledge, there is no computer vision system which can possibly come up with a 'correct' interpretation in a couple of seconds.



(a) (b)

Figure 1.1: (a) Lake scenery (b) A part of the 'lake' scene (in a different resolution).

While the central nervous system may not be the only information processing system that can accomplish seeing so successfully, it is nevertheless the most robust and versatile system known to humans. It is thus not surprising that much progress in computer vision and image processing is the direct result of borrowing the properties of the HVS. For example, the use of the 'masking effect' of the HVS improves the effectiveness of image compression and restoration algorithms [71] [4] [16].

Naturally, there are also direct attempts in building up biological-like networks (artificial neural networks) to solve some image processing and vision problems [141] [40] [87] [135]. Although such networks are still crude, they have achieved some success recently. This work will focus on the plausibility of using a class of artificial neural networks for edge detection and image restoration.

1.2 The Nature of Images

An image of a scene is formed by projecting part of the three dimensional (3-D) material world onto a two dimensional (2-D) plane. Hence the image records certain properties of the material world, which are responsible for the nature of images. The 3-D world consists of objects which tend to exhibit some degree of coherence in a range of appropriate scales (ie. hierarchical organization), as a consequence of the cohesiveness of matter. This entails that the image consists of a background and images of objects which appear to be near constant or slow varying intensity regions under appropriate lighting conditions, and that objects appear to be more or less similar when inspected over a range of scales (see figure 1.1, the sail and the trees appear similar at two scales).

This property that the material world is hierarchically organized also implies that objects in images occur over a range of scales and that a natural image more or less exhibits a similar structure and appearance (in a general sense) whatever the area under inspection.

Apparently, given an appropriate scale, some objects in an image will appear large and their boundaries can be approximated by lines which are characterized by their positions and orientations. This implies that to extract boundaries of objects adequately, the scale should be take into account. In other words, features in a given scale will respond maximally to a filter with the corresponding window size. This suggests a multiresolution(or multiscale) analysis, which will be examined more closely in chapter 2.

As the properties of an image are the results of the underlying physical laws, any image processing system will have to utilize them to extract the information about the corresponding part of the 3-D world. It is thus not surprising that physiological and psychophysical experiments showed that the mammalian visual system has visual cells tuned for particular orientation and scales in the retinal image [54] [55]. It would be expected that if an image model is of some utility, it should also exhibit such properties.

1.3 Noise, Ill-posedness and Uncertainty

The purpose of image processing techniques such as image restoration is to recover information from 2-D images; that is, they are inverse problems. As an image is typically formed by projecting a 3-D scene on a 2-D image plane, a consequence is the loss of information in the process of forming the image. This implies that an image may correspond to an infinite number of possible scenes. Such problems are ill-posed, in the sense that they do not have unique solutions [104]. A common approach to these problems is to assume that they do not require domain-dependent knowledge, but only generic constraints about the physical world, such as continuity. Such an approach is known as regularization theory. This notion will be discussed in more detail in Section 3.5.

Another form of ill-posedness is caused by the fact that the image formed is inevitably corrupted with random noise and sometimes blurred. A common type of noise can be modelled as Gaussian white noise, and can be reduced by a smoothing operation. To reduce the effect of blurring, which can be viewed as a smoothing operation, a common approach is to apply an inverse operation. However, the inverse operation is in general a kind of highpass filter which will amplify broadband noise as well; in other words, a small amount noise or error in the data can produce an extremely large error in the solution. Thus, there is a conflict in the operations to reduce blurring and noise. This phenomenon is analogous to the uncertainty principle in vision, which is an inevitable consequence of the inherently contradictory nature of the requirements of vision – an inference process concerning the recognition of 'what' is 'where' [129] [131] [130], for example, in image segmentation, the trade-off between the resolutions of the class space and the position space [131]. Instances of the uncertainty principle will be given in chapter 2 and of deblurring in chapter 5.

1.4 Image Modelling and Optimization Techniques

To deal with the image processing problems effectively using a minimal set of assumptions, it is necessary to use statistical methods, which represent image structure probabilistically rather than in terms of 'real world' constraints. In this approach to image processing, usually digital images are regarded as samples from a 2-D grid of random variables (pixels). The set of random variables is usually as large as 512×512 and 1024×1024 is not uncommon. To process such large amounts of data efficiently, it is preferable to have an underlying model which explains the statistical characteristics of the given data. In other words, if an image model is to be of use, it should have the ability to render the properties of images, that is, both to describe images observed and to generate synthetic images from model parameters.

As noted in section 1.2, a natural image usually consists of a collection of more or less constant areas. It is therefore reasonable to assume that the grey-level at a pixel is statistically dependent on some of its neighbours. Indeed, the use of neighbourhoods is inescapable when dealing with ambiguity and noise in image processing. It provides a geometric framework where pixels are classified using their *context* or neighbourbood. Based on this, ad hoc and heuristic methods, such as relaxation labelling, have been developed for edge detection and segmentation [142] [41] [137]. The main idea in relaxation labelling is to define 'compatibility functions' which are much like statistical correlations and are used to quantify local constraints. The process usually is implemented as a network of local computations which are intended to be homogeneous and can be performed in parallel. This approach is similar to the edge detection method used in this work (chapter 4).

Let $\{x_{ij}, 0 \leq i,j < M\}$ be the grey-level image, of size M by M pixels. One way to describe the statistical dependence among neighbouring pixels in an image is to represent x_{ij} as a linear weighted combination of pixels in the neighbourhood of x_{ij}. Using the assumption that images are wide sense stationary, fast algorithms based on this class of image models have been developed for image restoration, edge detection and image compression [60] [61]. The estimates of the model parameters are usually obtained from training data using a minimum mean square error (MMSE) or Maximum Likelihood method. However, the lack of a facility to describe scale in this class of models motivated the development of a class of multiresolution linear models. This will be discussed in chapter 2.

Alternatively, the local dependence characteristics of an image are formulated using the concept of Markovian random fields(MRF) [133]. Formally, an image is said to be a Markovian random field if its conditional density $p(\cdot|\cdot)$ satisfies

$$p(x_{ij}|x_{st}, {\scriptstyle 0\leq s,t<M,(s,t)\in\mathcal{N}\wedge(s,t)\neq(i,j)}) = p(x_{ij}|x_{st}, {\scriptstyle (s,t)} \in \mathcal{N} \wedge {\scriptstyle (s,t)} \neq {\scriptstyle (i,j)}) \qquad (1.1)$$

where $\mathcal{N}$ is the neighbourhood of (i,j) which determines the order of the MRF.

Markovian random fields have been applied to image restoration and edge detection [35] [9] [34] [125]. In principle, such an approach enables the two different problems to be combined under the framework of Bayesian inference [35]. An obvious criterion for the choice of the solution of such problems is probability of error, which gives as a solution that having overall maximum probability, given the image data. More specifically, given the set of observations $Y = \{y_{00}..y_{ij}..y_{mm}\}$, the *maximum a posteriori* (MAP) estimate $\hat{X}$ of the true image $X = \{x_{00}..x_{ij}..x_{mm}\}$ is the vector $\hat{X}$ which maximizes the joint *a posteriori* probability distribution as follows

$$\hat{X} = \arg(\max_{\tilde{X}} P(\tilde{X}|Y) = P(Y|\tilde{X})P(\tilde{X})/P(Y)) \qquad (1.2)$$

As the denominator $P(Y)$ can be ignored in the maximization, only knowledge of $P(X)$ - the *a priori* probabilities of the label configurations and $P(Y|X)$ - the conditional probability distribution of the observation given the true image, must be known. They are usually assumed to be a certain distribution (eg. Gaussian distribution) and their parameters are estimated from training data (eg. [35] [9]). The global MAP estimate, however, is computationally demanding because of the need to search for the optimal estimate over all the possible configurations. To tackle this difficulty, various methods which decompose the global optimization into an iterative network of local computations have been introduced.

Perhaps the simplest way is to add more restrictions to design a subclass of MRF, for example Markov mesh models proposed by Abend *et al.* [2]. However, Markov mesh models have a causal Markovian dependence in the image plane. Hence they are not natural in a spatial context and can be too restrictive.

Using the theorem that an MRF with respect to a neighbourhood $\mathcal{N}$ is equivalent to the sample space of a Gibbs distribution, Geman and Geman proposed a stochastic relaxation algorithm for MAP image restoration [35]. This stochastic relaxation algorithm has its roots in simulated annealing and the Metropolis algorithm for optimization [9]. More specifically, it is an iterative application of the Metropolis algorithm with a gradually decreasing temperature (a procedure reminiscent of the cooling process for metal or chemical substances, hence the name simulated annealing). Stochastic relaxation will converge to the equilibrium distribution which gives the wanted configuration of minimum energy. This approach has inspired a few researchers to construct energy functions and cast image processing problems as energy minimisation problems using the simulated annealing technique (eg. edge detection [119], segmentation [74]).

The simulated annealing scheme is, however, notorious for its need for a slow temperature decrease, which usually requires hundreds of visits to each pixel. Because of this weakness, various methods for speeding convergence have been proposed. In solving combinatorial optimization problems, instead of simulating stochastic annealing, it is possible to estimate its mean behaviour using mean field theory [22]. The mean field annealing algorithm is a deterministic relaxation procedure which exhibits faster convergence while often preserving the quality of the solution afforded by the simulated annealing approach. There is, of course, no lack of applications of the mean field theory for edge detection [45] [139] [140].

Another optimization scheme, called Iterative Conditional Modes(ICM), was proposed by Besag [9], which is exactly equivalent to 'instantaneous' freezing in simulated annealing. Thus, it will not guarantee to find the MAP but it does converge to a local minimum. Nevertheless, ICM is appealing because of its fast convergence. In the application of grouping and closing gaps in edges, Urago *et al.* modelled edge images as MRFs and defined the Gibbs distribution associated with it [125]. They also introduced an energy function which reflects the structure of edges and used ICM relaxation to minimise the energy function.

There have also been attempts to accelerate the convergence of Geman's stochastic relaxation algorithm using multigrid techniques such as the *renormalisation group* technique [37]. In [37], Gidas proposed a Renormalisation Group (RG) method for image restoration. When the RG algorithm is combined with the simulated annealing technique, it is called Renormalisation Group simulated annealing(RGSA). The RGSA algorithm provides a multiscale, coarse-to-fine framework, which reduces the computation of processing by combining local processing at different scales with an interscale transfer of information.

In short, since image features may occur at various scales and because of the inherently contradictory nature of the requirements of vision, there is a need to combine global information and local information, so that a coherent solution can be obtained. The processing of combinations of global and local information can be cast as an optimization problem which can be solved efficiently by combining iterative and multiresolution techniques.

1.5 Neural Networks for Image Processing

The past decade has also seen various artificial neural networks applied to image processing problems [141] [27] [70] [8] [122]. An artificial neural network(ANN) is a network of processing nodes which are interconnected with adjustable, weighted links. A review of some generic models for image processing is presented in this section. Instead of considering the modelling of biological neural networks(cf. chapter 3), the purpose of this section is to highlight the significant features of ANNs for image processing.

One of the motivations for the application of artificial neural networks to image processing problems is the hope that an adaptive computer vision system can be built by adopting the architectures and algorithms used by the brain, whose ability in association, generalisation, classification, feature extraction and optimization has not been duplicated using conventional computational techniques. The brain is a hierarchical layered neural network and each layer operates in parallel. A class of artificial neural networks which captures these features is the feedforward networks (see figure 1.2). Given a set of training data, a feedforward neural network adjusts

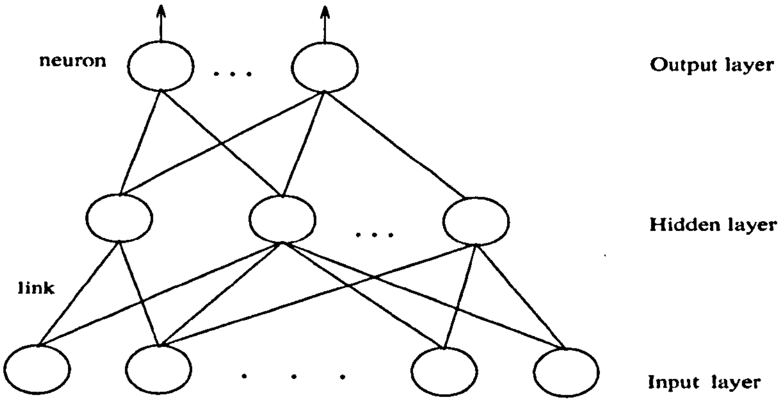


Figure 1.2: A feedforward network.

its link weights so that it implements a mapping which either matches the training data as closely as possible or transforms the training data into uncorrelated outputs. After training, the feedforward network is used to classify input data or decorrelate input data.

There are of course learning mechanisms based on different criteria (eg. the principle of maximum information preservation by Linsker [80] or the least mean squares (LMS) [112]) for the adaptive abilities of linear and nonlinear feedforward networks. Using the LMS criterion, Daugman suggested a three layered artificial neural network for transforming two dimensional images into generalised nonorthogonal 2-D Gabor representations for image analysis, segmentation and compression [27]. A single-layer linear feedforward neural network, proposed by Sanger [112] has weights which will converge to the first few eigenvectors of the autocorrelation matrix of the input data. Like the Karhunen-Loève transformation, this single-layer linear feedforward neural network was applied to some simple tasks in image coding, texture analysis and feature extraction [112]. Obviously, feedforward networks provide alternative methods

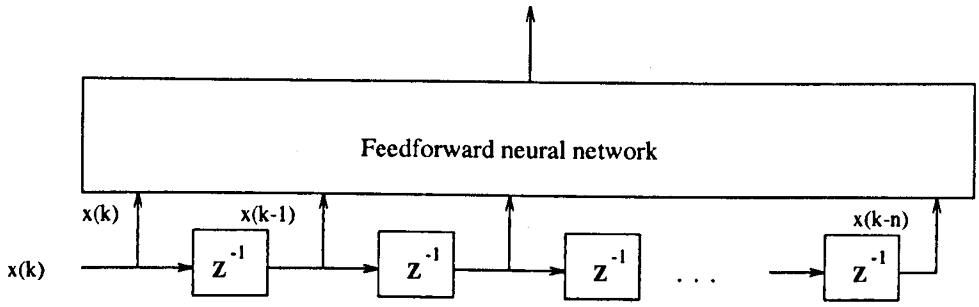


Figure 1.3: A time delay neural network.

for the implementation of simple image transformations.

Feedforward networks with only one hidden layer using any squashing functions have been proved to be able to approximate any Borel measurable function to any desired degree of accuracy [51]. It is not surprising that feedforward networks are used for nonlinear filtering [98], and pattern recognition problems [99]. Equipped with a tapped delay line, feedforward networks are also capable of modelling systems where the output has a finite temporal dependence on the input. Such an architecture is often referred as a Time Delay Neural Network (TDNN)(see figure 1.3). When the feedforward neural network is linear, this structure is equivalent to a linear finite impulse response filter (FIR). Applications of TDNNs to speech synthesis, recognition and nonlinear prediction were reported with good results [115] [75] [76].

Currently, much work on feedforward networks has concentrated on fast learning algorithms and on improving their generalisation abilities [57]. Compared with conventional image processing techniques, it seems that feedforward neural networks are attractive for their abilities in 'learning' and their comparatively simple architectures.

While feedback is common in the brain, there is no feedback mechanism in feed-

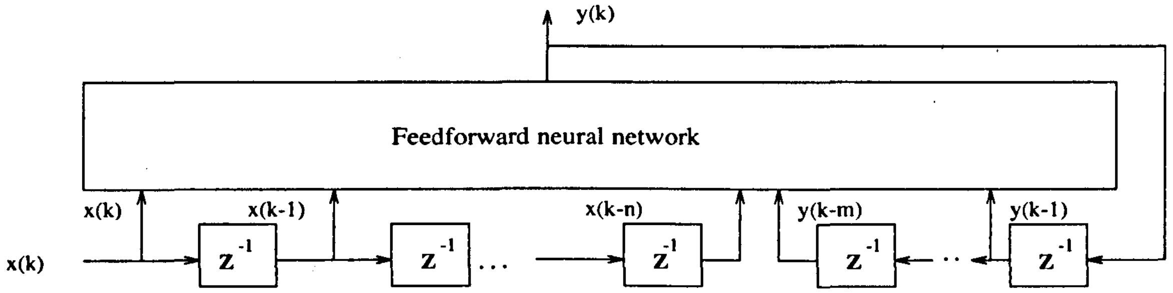


Figure 1.4: An output feedback neural network.

forward networks. Networks which possess feedback connections are called recurrent networks and are inherently recursive [57]. Recurrent networks fall into two categories: (a) output feedback and (b) state feedback. Output feedback recurrent networks have an architecture shown in figure 1.4. Apparently, when the feedforward neural network is linear, this structure is equivalent to a linear recursive filter. This class of output feedback networks has been applied to nonlinear system identification and control problems (eg. [95]).

The state feedback recurrent networks are generally single-layer networks with feedback connections between nodes (see figure 1.5). If any node can be viewed as input or output node, this class of networks is perhaps the most general. Intuitively, state feedback networks can be easily constructed for representing the interaction of competitive and cooperative constraints for solving image processing problems (eg. Grossberg used a competitive network for boundary grouping[40]).

The most popular state feedback network for image processing problems is probably the Hopfield network. Hopfield networks have been applied to image restoration [141], segmentation [137] and edge detection [135]. Details of the Hopfield type of net-

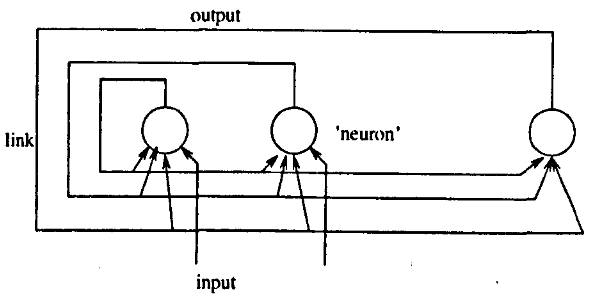


Figure 1.5: A state feedback neural network.

works will be presented in chapter 3 and a hierarchical Hopfield network is proposed for edge detection and image restoration.

A generalisation of the Hopfield network is the Boltzmann neural network, which uses simulated annealing for locating the global minimum of its energy function. It is similar in spirit to the stochastic relaxation algorithm of Geman and Geman [35]. While the stochastic relaxation is in the same world as the Boltzmann neural model, the ICM is related to the Hopfield network model. Hopfield and Boltzmann networks therefore have the added attraction of a clear connection to the stochastic models which underlie most rigorous solutions to image processing problems. And there is no lack of applications of the Boltzmann neural model, for example, Sejnowski and Hinton [114] reported its use for separating figures from background.

1.6 Requirements for Image Models

From the discussion of the previous sections, there are several issues involved in solving general image processing problems. As natural images consist of more or less slowly varying grey level regions, an effective image model must have a mechanism for expressing smoothness and discontinuity. The fact that the world is hierarchically organized entails that the underlying model should be able to render objects of different scales as well. Equally important, a facility is needed to cope with the uncertainty principle in vision. Finally, because of different distortions and information lost in the process of forming images, *a priori* knowledge may be required to recover the information.

This thesis examines the use of artificial neural networks for edge detection and image restoration. For this purpose, there are several requirements that will be considered next.

1.6.1 Requirements of Edge detection

A simple approach to edge detection is to apply a pair of highpass (gradient) filters, followed by an edge connection process (eg. [89]). However, when noise is present, problems arise. An effective algorithm for edge detection must address the following requirements:

1. Local processing/high position resolution: edges are the places where image properties change abruptly. Thus, it is required to have a high position resolution for edges. This implies that a small size of window is required. Apparently, edge detection is an intrinsically local operation, which may enable computation in parallel.

2. Noise immunity: when noise is present, the highpass filtering operation will amplify it. It is thus necessary to use a larger window for reducing the effect of

noise. The requirements of noise immunity and the high position resolution are incompatible and need a scheme for trade-off.

3. Boundary continuity: in grouping edge pixels, it is necessary to assume that boundaries of objects are piecewise continuous plane curves, ie. they do not contain too many corners.

4. Flexibility: When a priori knowledge is available, the algorithm should be readily extended to incorporate it for better performance.

1.6.2 Requirements of Restoration

For restoration of images distorted by known blurring and noise, an effective algorithm has to consider the following issues :

1. Ill-posedness/regularization: Since the blurring usually is in an ill-conditioned matrix form, its inverse may not exist or may be error-sensitive. Thus a direct application of the blur matrix inverse may not be possible. Certain assumptions are needed to regularize the solution space.

2. Edge enhancement: Edge information is so important in perception that to a human, an image with sharper edges but a lower signal noise ratio are usually more pleasant than an image with blurred edges but a higher signal noise ratio. Thus, identification of edges should be included for edge enhancement.

3. Noise smoothing: In smooth areas, noise is much more noticeable and may result spurious edges. Hence, it is necessary to filter noise in slowing varying or constant

grey level areas

4. Flexibility: When a priori knowledge is available, the algorithm should be able to incorporate it for better performance, but the results should be robust, ie. not too sensitive to small changes in parameters.

A great deal of recent work has shown how image data and world knowledge can be integrated within the framework of minimisation of energy functions [104] [120]. If it is possible to design an energy function which reflects the quality of solutions, then a Hopfield network can be employed to find a solution.

1.7 Thesis Outline

This work attempts to demonstrate that the combination of a multiresolution model and regularization theory can provide an effective framework for image processing and a class of image processing problems can be solved using a hierarchical Hopfield network, which is an effective implementation of this model.

This thesis is organized as follows: in chapter 2, stochastic modelling of images is discussed. In particular, a class of linear multiresolution image models will be introduced. It is emphasized that a natural way to deal with the uncertainty problem is the use of a multiresolution representation. This chapter is concluded with a multiresolution iterative model for image processing.

Chapter 3 is an introduction to the properties of neural networks, which are relevant to edge detection. An interesting observation is that the HVS uses a priori knowledge for vision tasks. This is the basis on which regularization theory is built. A hierarchical Hopfield network is proposed to implement the multiresolution iterative model of chapter 2, which encapsulates the iterative and smoothness regularization ideas.

Chapter 4 will detail an algorithm for edge detection using the proposed hierarchical neural network. The idea is to specify the spatial relations of edge properties of pixels in terms of energy. When they correspond to an edge, the energy term will decrease. The energy function is minimized by using a Hopfield network which converges to the edge map. Results of applying this scheme to a variety of test images are presented and discussed.

It is shown that a class of image restoration problems can be regarded as energy minimization problems in chapter 5. Again, a cost function whose minimisation will reflect the desirable solution is specified as the energy function of the proposed hierarchical Hopfield network. An adaptive restoration scheme consisting of the edge detection method proposed in chapter 4 and an adaptive filtering method is proposed and results of tests are presented. The implications of these tests for the approach used are discussed.

Finally, a summary and the contribution of this work is presented in chapter 6. A discussion of the limitations of the model and the proposed hierarchical Hopfield neural network is also summarised.

Chapter 2

Multiresolution Image Models

2.1 Introduction

Image models give a quantitative description of images whose formation is prone to noise and distortion. Over the past three decades, researchers have proposed different image models in order to provide a framework within which various image processing techniques can be considered and analyzed, hence enhanced. As an example, Wiener filters in restoration problems are derived from a stationary statistical model which characterizes images by their covariance functions[61]. To permit characterization of local properties of pixels, a linear system describing the relationship between a group of pixels by a difference equation and forced by white noise with a known power spectrum is a useful approach for representing the ensemble of images. Depending on their pixel neighbourhoods, such models are categorized into three groups, namely causal, semicausal and noncausal models.

In this chapter, a description of stochastic image models is presented. Emphasis is put on noncausal multiresolution models, which have been successfully applied to various areas in image processing and computer vision, (eg., feature extraction[18], data compression [86][121]). Particularly, a multiresolution model which is a generalized model from [23] is introduced for edge detection and image restoration.

2.2 Stochastic Image Models

The stochastic modelling of images treats the spatial coordinates of an image as time-like indices. Modelling images as stochastic processes is not new: there are many models found in such areas as data compression [61]. An example is a simple 1-D causal autoregressive(AR) model, borrowed from work on 1-D signals such as speech and applied to images via a raster line scan, which is found in most image capture and display systems. Based on AR models, the differential pulse code modulation(DPCM) coding methods are used in data transmission applications [61]. A DPCM method works because in general adjacent pixels in an image have high correlation. Models which can capture the correlation property of images may give huge reductions in the volume of information which it is required to transmit.

Although 1-D causal models are simple and they have had a degree of success, such line-by-line methods force an essentially 2-D image into 1-D signals, so that those spatial structures inherent in the image are difficult to take into account. In applications where whole images are available, there is no good reason to impose causality artificially and it is natural to consider other data structures and models to characterize 2-D images.

Among other methods, Markov random fields and 2-D noncausal prediction models are general image models which can reflect the local properties of images [35][61]. However, they may demand more computation. Recently, a class of multiresolution image models which combine the simplicity of an AR model and the effectiveness of different scale features description have emerged to give efficient solutions to many image processing problems. This section will be an account of image models as the prelude to a multiresolution model. A multiresolution model which combines a propagation process and a lateral interaction process will be introduced and used as the framework for edge detection and image restoration.

2.2.1 One Dimensional Image Models

An autoregressive process of order p is a zero mean random process $x(l)$ which can be generated by equation (2.1). Let $x(l)$ be the signal and $\omega(l)$ be the innovation term which is assumed to be a sample of white noise, then a stationary pth order AR model is given by

$$x(l) = \sum_{i=1}^{p} \alpha_i x(l-i) + \omega(l) \tag{2.1}$$

The AR models are sometimes called causal minimum variance representation (MVR), for the signal estimator

$$\hat{x}(l) = \sum_{i=1}^{p} \alpha_i x(l-i) \tag{2.2}$$

is the linear estimator which gives minimum mean square prediction error[61]. Given $x(l)$, α_i can be determined by the Yule-Walker equations. It can be also solved by Levinson's algorithm which is in a recursive form and is easy to implement[101]. Apparently, by introducing a scan the above model can be used as a model for images, but the introduction of a scan path is not fully justified in many image processing applications.

2.2.2 Two Dimensional Image models

Features in an image are characterized by their spatial structures, which cannot be exploited by the line-by-line models mentioned in the previous section. For example, in an edge area, pixels which are scanned in the direction of the edge are highly correlated, but when scanned across the edge they are less correlated. Thus an edge not along the scan direction will be treated uncorrelated but in reality it is correlated. Naturally, a straightforward approach to model 2-D image is to generalize the 1-D stochastic signal models [61]. Thus, an image is expressed by a 2-D stochastic process, or *random field* and the image is the output of a linear system driven by noise, as

$$x_{ij} = \hat{x}_{ij} + \omega_{ij} = \sum_{mn\in\mathcal{N}} a_{mn} x_{i-m,j-n} + \omega_{ij} \tag{2.3}$$

where $\{x_{ij}\}$ is the random field,$\{\hat{x}_{ij}\}$ is the linear estimator of $\{x_{ij}\}$ and $\mathcal{N}$ denotes a neighborhood of (i,j). There are three types of linear prediction models, according to the definition of $\mathcal{N}$, namely,

$$\mathcal{N} = \begin{cases} \{-p \leq m \leq p, 1 \leq n \leq q\} \cup \{1 \leq m \leq p, n = 0\} & \text{causal} \\ \{-p \leq m \leq p, 0 \leq n \leq q, (m,n) \neq (0,0)\} & \text{semicausal} \\ \{-p \leq m \leq p, -q \leq n \leq q, (m,n) \neq (0,0)\} & \text{noncausal} \end{cases} \tag{2.4}$$

If $\{\omega_{ij}\}$ denotes for the prediction error and $\{\hat{x}_{ij}\}$ is set to be a minimum variance predictor, the representation is a MVR which is equivalent to a Bayesian estimate in a Gaussian Markovian random field[133].

Although these 2-D models do take account of inter-line correlation, it is not hard to see that the causal and semicausal models are biased by their scanning directions. And all the three types of model lack the facility to take account of features of different scales, even though a noncausal model combined with iterative techniques, such as relaxation methods, can capture global features [35]. Nevertheless, such approaches are usually slow and expensive in computation. Next, a new class of general linear models will be presented. The models are causal in scale space and non-causal in the two dimensional spatial plane. This allows a recursive estimator to be implemented[24].

2.3 Towards Multiresolution Image Models

2.3.1 Motivation

In the 1940s, Gabor in his classic paper on audio analysis pointed out that the traditional Fourier frequency decomposition of a signal is limited by a signal property known as the uncertainty principle[32]. In a different context, Wilson and Granlund used the uncertainty principle to explain some inherent properties in vision [129]. Later, Wilson and Spann used a similar argument in discussing the problem of segmentation, which is illustrated in the following.

A basic question in image processing and computer vision is to estimate 'what' is 'where' in the image[88]. A fundamental problem in this field is edge and boundary detection, in which the boundary properties(eg. orientation) are 'what', while their location is 'where'. A simple solution to the edge detection problem is to convolve the image with a set of fixed sized filters and apply a threshold to a function of the set of convolved images. Such an approach leads to a less satisfactory result because the features we perceive and find meaningful vary enormously in a range of scales [88]. Furthermore, when the input image is corrupted with white noise, the above solution will be riddled with false edges and missing edges. To deal with such cases, the image is first smoothed, then the set of kernels is applied. Although the effects of noise can be reduced, a new problem arises: the positions of the edges are 'blurred'. The more the image is smoothed, the fewer spurious edges will appear. Unfortunately, the certainty of the edge positions is reduced correspondingly (see also Canny[20]) . It is clear this dilemma is analogous to the uncertainty principle [131].

Wilson and Spann pointed out this trade off between 'spatial' resolution and 'class space' resolution and proposed the use of a multiresolution approach to overcome this difficulty[131]. The multiresolution model of Wilson and Spann is based on a quadtree structure(see Figure 2.1). They used an iterative clustering method to decide the classes in an image and a coarse-to-fine boundary estimation scheme.

Another benefit of multiresolution models is their ability to represent image features over a range of scales. For example, in the pyramid representation, an image is described by coarse features and progressively refined till the image plane is reached. In this context, data compression is possible [129] [17]. This coarse-to-fine recon-

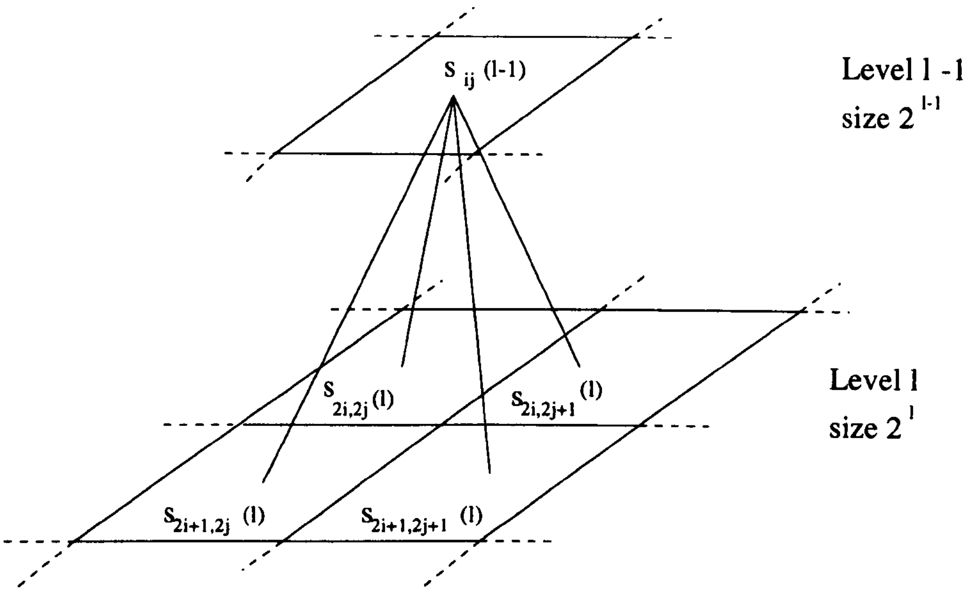


Figure 2.1: A quadtree structure

struction process is the very property which underlies a class of linear multiresolution models [24]. Also in the coarse-to-fine reconstruction, the features in the coarse resolution intuitively can be regarded to be accurate with high confidence. Thus, these features can be used to control or '*regularize*' the reconstruction of features in the finer resolution [135] [37].

2.3.2 Multiresolution Representations

In a multiresolution representation, different scales give similar but different descriptions of an image. Thus, if such representations are to be useful, the problem of combining information across scales must be dealt with effectively. Before discussing this 'scale consistency' problem[132], it is perhaps worth giving a brief description of various multiresolution representations.

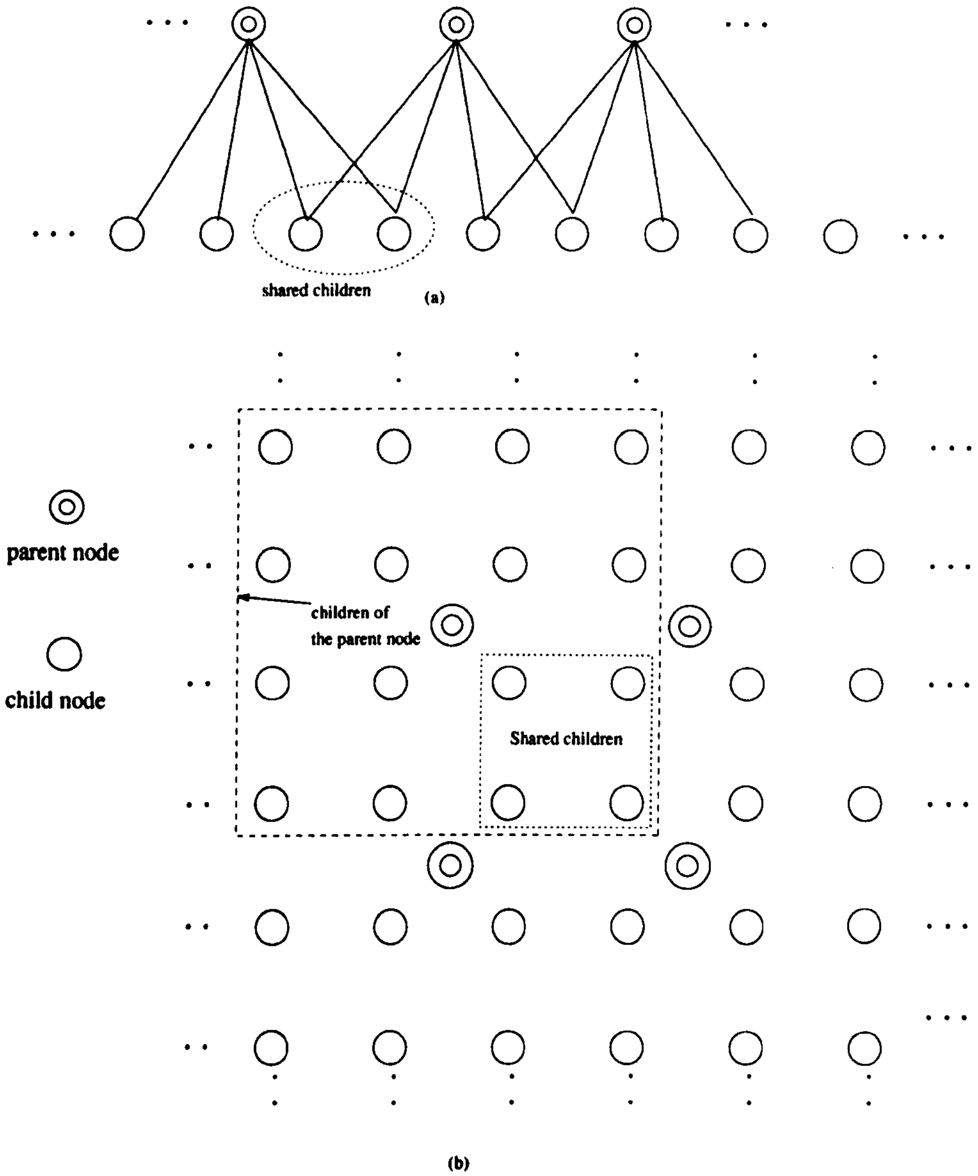


Figure 2.2: (a) A pyramid tree structure for 1-D signal (b) its 2-D counterpart.

1. Scale space

This representation was proposed by Witkin[132]. In 1-D, the space-scale representation of a signal $f(x)$ is constructed by smoothing the signal with a family of functions of a continuous scale parameter σ, such as Gaussian functions $g(x, \sigma)$ of increasing width, so that the scale-space representation $F(x, \sigma)$ of the signal is also a function of σ, which can be written as

$$F(x, \sigma) = f(x) * g(x, \sigma) \tag{2.5}$$

where $*$ is a convolution operator. This family of derived images can also be viewed as the solution of the heat diffusion equation with the initial condition

$$F(x, 0) = f(x) \tag{2.6}$$

Thus, the techniques used in the study of heat diffusion can be readily borrowed [103] [68]. Koenderick [68] and Lindeberg[79] have further developed the theory of scale space representations and the generalization to 2-D is straightforward.

2. Pyramids

A pyramid structure of an image is a tree constructed by smoothing the image and then subsampling the smoothed image. This operation is carried on recursively until there is one node, called the root left. Usually the scale space axis is logarithmic (base 2) so that the size of each level in the pyramid is 4 times that of its parent level. About 30% extra memory space is needed to construct the pyramid. Because of the smoothing and decimation operations, each level in the pyramid represents a different resolution of the image. The pyramid structure can be written

$$\begin{aligned} f_{ij}(M) &= f_{ij} \\ f_{ij}(l-1) &= \sum_{mn} A_{ijmn}(l) f_{mn}(l) \end{aligned} \tag{2.7}$$

where $f_{ij}(l)$ is the image data at level l and $A_{ijmn}(l)$ represents a filter. Note that a child node $f_{mn}(l)$ can have several parents. Figure 2.2 is an example of a pyramid structure. A special case of the pyramid is a quadtree structure, in which

$$A_{ijmn}(l) = \begin{cases} 1/4 & \text{if } (m,n) \in \{(2i+s, 2j+t), s = 0,1, t = 0,1\} \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

In other words, each child node has only one parent node (see figure 2.1). More explicitly, the quadtree is constructed using the following equation

$$f_{ij}(l-1) = \frac{1}{4}(f_{2i,2j}(l) + f_{2i,2j+1}(l) + f_{2i+1,2j}(l)) + f_{2i+1,2j+1}(l)) \tag{2.9}$$

3. Wavelet Transforms

While a pyramid structure needs about 30% extra memory to represent an image, an orthonormal wavelet transform of an image will need exactly the same amount of memory space to represent the image. A wavelet transform representation is also constructed by successive filtering and decimation operations. A 1-D wavelet transform representation of a 1-D signal f_i is given as follows,

$$f_i(l) = \sum_k h_{2i-k} f_k(l+1) \tag{2.10}$$

$$d_i(l) = \sum_k g_{2i-k} f_k(l+1) \tag{2.11}$$

where $h(\cdot)$ and $g(\cdot)$ form a conjugate mirror filter pair [26] The reconstruction of the signal f_i from its wavelet transform representation$\{d_i(l)\}$ is simply

$$f_i(l+1) = \sum_k h_{2k-i} f_k(l) + \sum_k g_{2k-i} d_k(l) \tag{2.12}$$

An important feature of such wavelet representations is that the set of wavelets used is orthonormal. The set of wavelets, is partially dilation and shift invariant, in other words, each member can be obtained by dilating and shifting another member in the basis. The 2-D wavelet transform is generalized from the 1-D case (cf. [86]), normally by using a Cartesian separable transform.

The above multiresolution representations are all designed to deal with the uncertainty problem in signal and image processing and obvious properties of images, such as structures spanning a range of scales. The pyramid structure also leads to effective image modelling. The pyramid structure was, perhaps, first used by Rosenfeld and Thurston for edge detection [110]. Although the orthonormality property of the wavelets leads to compact representations and efficient decomposition of an image, it only provides a partial solution to the uncertainty problem. More sophisticated structures, such as Multiresolution Fourier Transform(MFT) have been proposed for image analysis [18] and texture segmentation [52].

2.3.3 Linear Multiresolution Models

Among the pioneering works of stochastic multiresolution image modeling, perhaps Clippingdale and Wilson [24] were the first who considered the scale index to be time-like and derived an AR model in the scale space. Their model is a special case of a generalized linear multiresolution model which is also proposed by Clippingdale [23]. The same stochastic modelling in scale space is also used by Basseville *et al.* [6] in the wavelet transform representation. There are also works which treat the scale space as a Markov chain [10] and use the MAP(usually nonlinear) criterion to combine information between successive levels [37].

The class of general linear multiresolution models proposed by Clippingdale can be recursively defined in a matrix notation as

$$\begin{aligned} S(0) &= \mathbf{B}(0)W(0) \\ S(l) &= \mathbf{A}(l)S(l-1)+\mathbf{B}(l)W(l), 1 < l \leq M \end{aligned} \tag{2.13}$$

where l is the level index, $S(l)$ and $W(l)$ are the lexicographically ordered vector representations of the signal and the innovation at level l respectively, $\mathbf{A}(l)$ and $\mathbf{B}(l)$ are filter matrices. In image processing, writing the linear operator explicitly, it takes the form

$$s_{ij}(l) = \sum_{mn} A_{ijmn}(l)s_{mn}(l-1) + \sum_{pq} B_{ijpq}(l)w_{pq}(l) \tag{2.14}$$

where $s_{ij}(l)$ is the image (or features) at level l and w_{pq} is the innovation term. The initial condition is

$$s_{00}(0) = B_{0000}(0)w_{00}(0) \tag{2.15}$$

Notice the equation is similar to equation (2.1), and thus a MMSE estimator can be derived using the well-established techniques for state-space methods[101]. Using this approach, given the set of noisy data,

$$\tilde{s}_{ij}(l) = s_{ij} + \nu_{ij} \tag{2.16}$$

an estimator for this general linear model is expressed by a linear combination of data on the current level and the vertical propagation, or 'prediction' as follows

$$\hat{s}_{ij}(l) = \sum_{mn} A_{ijmn}(l)\hat{s}_{mn}(l-1) + (1 - \alpha_{ij}(l))\tilde{s}_{ij} \tag{2.17}$$

where $\alpha_{ij}(l)$ is a combination coefficient to trade off the reliability of data and the propagation 'prediction'.

A special case of the general linear multiresolution model is based on the quadtree representation, which takes the form

$$s_{ij}(l) = s_{\frac{i}{2}\frac{j}{2}}(l-1) + w_{ij} \tag{2.18}$$

One of the distinguishing characters of this model is that the image is tessellated into inhomogeneous blocks of different sizes. The implication is that each region (block) represents one feature; in other words, various local image structures are represented at different scales in this model. A scheme to model curves based on the

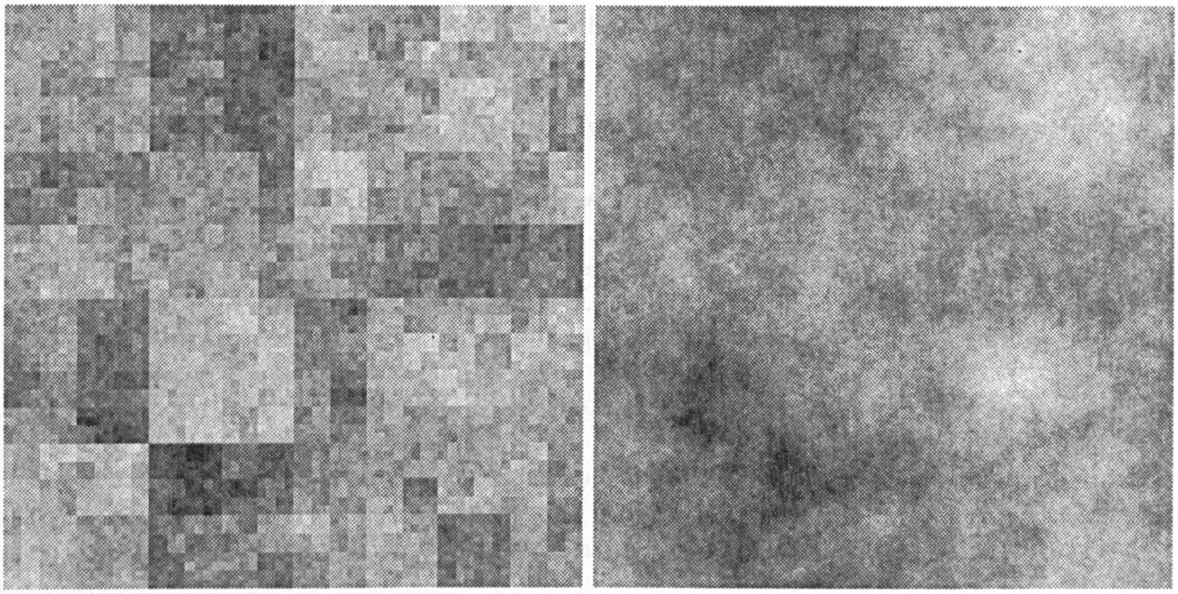


Figure 2.3: (a) A realization of a quadtree model and (b) a realization of a pyramid model.

quadtree model can be found in [18]. It is not hard to see that such a model leads to a simple structure and when applied to estimation problems, an exact minimum mean square error estimator can be derived using the same techniques from 1-D AR models (see Clippingdale and Wilson [24]). However, the very non-overlapping tessellation property of the quadtree model causes unwanted blocking effects which must be overcome. Such a problem can be overcome by allowing a child to be shared, i.e. a pyramid model,

$$s_{i,j}(l) = \sum_{mn} A_{ijmn}(l) s_{\frac{i}{2}+m,\frac{j}{2}+n}(l-1) + w_{i,j} \tag{2.19}$$

Figure 2.2 is a pyramid structure first used by Rosenfeld and Thurston for edge detection [110]. In figure 2.3, images generated by quadtree and pyramid models are shown. Comparing figure 2.3(a) with figure 2.3(b), a pyramid model is more natural, although its vertical propagation requires more computations.

Following Clippingdale and Wilson's work based on a quadtree representation, a variety of linear multiresolution models in different representations are used in estimation [23], coding [121], segmentation [10] and curve extraction problems [18] [28]. Different representations utilize different strategies which are realized by $A_{ijmn}(l)$ and $B_{ijpq}(l)$. Apparently, the coarse-to-fine refinement procedure in this general linear multiresolution model is an effective method to combine local and global information. In addition, it has the ability to render features of different scales.

2.4 A Processing Model for Image Restoration

2.4.1 Iterative Methods

A given signal model may be more or less effective at capturing image structure,but it will only be useful if it leads to computationally practicable solutions to problems. This aspect must now be considered. As noted in Chapter 1, many image processing problems can be formulated as optimization problems, so that the solution

$$E(x) = \min_{\{x' \in \Omega\}} E(x') \tag{2.20}$$

where $E(\cdot)$ is a cost function or an *energy* function related to the quality of the solutions and Ω is the solution space. Often, a direct computation for the optimal solution of a problem involves a huge matrix inverse operation, so that it is impractical. In search of the optimal solution, an alternative is to use local iterative methods, such as gradient descent methods, based on the MSE criterion or relaxation using a MAP criterion [35], in which costs or energies are minimized at each pixel on each iteration. This greatly reduces the computational cost and because of the local updating schemes, it is easy to use different strategies depending on local properties of the image, ie. adaptive methods. Another benefit of this local updating is a possible parallel implementation. However, a drawback is that local updating may lead to a local minimum only. Much effort has been devoted to overcome this drawback. One interesting approach is to use probabilistic schemes, such as stochastic relaxation algorithms (simulated annealing). Such an approach has now been widely applied to various image processing problems[35][34].

Because images are often formed by projecting 3-D scenes on 2-D planes, and usually corrupted by some noise and distortion, many image processing problems are ill-posed (see section 3.4). One way to solve an ill-posed problem is to define a criterion to select an approximate solution from a set of admissible solutions. This criterion so defined is called a *regularization* term, which reflects prior knowledge of the problem. The solution to the problem can then be rewritten as

$$\min_{\{x' \in \Omega\}} (E(x') + \alpha E_s(x')) \tag{2.21}$$

where $E_s(\cdot)$ is the regularization term. Using an iterative method, it is possible to control α according to local properties, such as edges [65]. Another way of incorporating prior knowledge is to impose constraints, so that only feasible solutions are accepted, eg. as the method of projection onto convex sets (POCS) does [65]. Both regularization and POCS can be easily embodied in iterative methods. The ideas of adaptation and regularization are taken to form a multiresolution model for image restoration which will be described next

2.4.2 A Multiresolution Iterative Restoration Model

The general linear multiresolution models described in section 2.3 can be used for edge detection and image restoration. The pyramid structure mentioned is used as it avoids difficulties, such as a spatial difference operator missing edges which lie on the borders of adjacent blocks [42].

The adaptive restoration algorithm consists of two stages: (1) an edge detection process (2) the restoration process. Both processes start from a certain low resolution in which noise is sufficiently reduced due to the lowpass nature of the pyramid representation. For each resolution, a lateral relaxation scheme is used to enhance long range characteristics. The obtained solution in coarser resolutions is propagated down and used to regularize the search of the solution space in finer resolutions. In short, there are three steps for both processes:(1) construction of input pyramids (2) lateral relaxation (3) vertical propagation.

1. Construction of input pyramids

To outline the algorithm, the first task in hand is to clarify what are represented by the models so that pyramids can be built. Since an edge controlled restoration scheme is desired, the model must be able to represent boundaries, orientation and gray level structures effectively. Three pyramids are built; they are (a) the gray level pyramid of the input image, (b) its orientation pyramid and (c) its edge map pyramid. The grey level pyramid is first built as the input for the restoration algorithm. The lowpass kernel to construct the pyramid should be isotropic to avoid introducing bias in orientation which is important characteristic of an edge and is used for edge determination[127] [135]. The orientation pyramid is constructed next by applying a pair of odd size kernels to the grey level pyramid. The orientation representation is double angled which is detailed in chapter 4. Finally, the edge map pyramid is obtained using a Hopfield network also detailed in chapter 4. The edge representation is similar to Bhalerao's work[10], but the difference is that edges in the model are represented by a set of edge pixels,

$$X(l) = \{x_{ij}(l) | (i,j) \text{ is a vertex site}, 0 \leq l \leq M\} \tag{2.22}$$

rather than the polygon model used by Bhalerao; in other words, a binary edge map is used.

2. Lateral relaxation

Many estimates and decisions in image processing problems are the results of interaction of many simultaneous local mutual constraints. To give a coherent solution, iterative methods outlined in section 2.4.1 are used to update a set of hypotheses (eg. there is an edge in location (i,j)). The local constraints are formulated as excitatory and inhibitory weights and the local decision (for edge detection) and estimates (for restoration) are made using the sum of all the local weighted constraints so that a globally consistent solution can be reached after a small number of iterations. Both lateral relaxations used in the edge detection process and the restoration process use odd size windows to gather local clues. The minimization of a cost or energy function is the basis of the two processes. The best fit to the data is sought within the priori knowledge and constraints.

3. Vertical propagation

The uncertainty principle asserts the trade-off between 'what' and 'where' [129]. Thus, the estimate obtained in the coarser resolution of the pyramid, where noise has been reduced by pyramidal smoothing, will be more reliable but its spatial resolution is low. To recover the spatial solution while maintaining the estimation reliability, the estimates in coarser resolutions are recursively propagated down using the stochastic linear multiresolution model expressed by equation (2.13). The propagation is achieved by a linear combination of the estimate obtained using lateral relaxation and the data in the immediately finer resolution. The linear combination is an average which has the effect that consistent features will be emphasized while noise which is uncorrelated will be reduced.

Figure 2.4 is a summary of the scheme. Note that information flows vertically and horizontally in this model which is reminiscent of the HVS (cf. Chapter 3) A full description of the edge process will be given in the chapter 4, and the restoration in chapter 5.

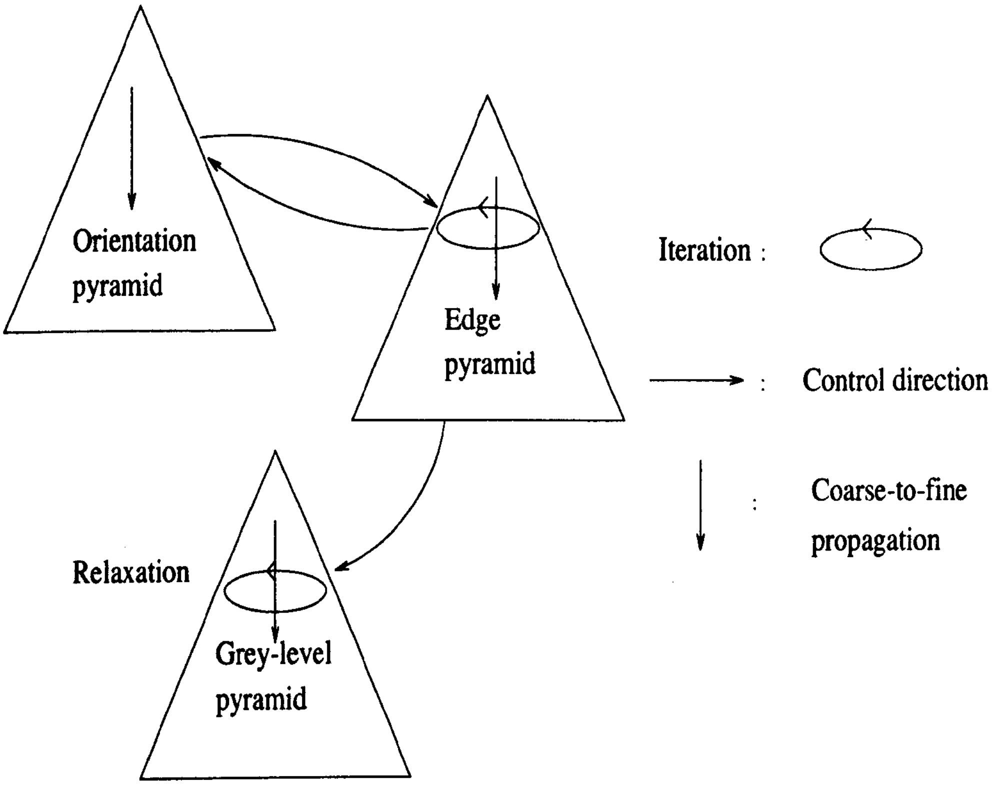


Figure 2.4: The restoration scheme.

2.5 Summary

The past decade has seen an ever growing interest in image multiresolution representation and modelling. This chapter has given a brief account of this development and proposed a framework which is general enough for various image processing problems. Its application to an adaptive image restoration was outlined. This framework combines the idea of regularization and iterative refinement based on a linear multiresolution model which includes multiple feature prototypes. It consists of two processes: a coarse-to-fine process and a lateral iterative optimization process, which will be further detailed when applied to practical edge detection and image restoration problems.

Chapter 3

Biological and Artificial Neural Networks

3.1 Introduction

It is interesting to know why a collection of much slower neurons is faster than a modern 'number crunching' digital computer, in particular when it comes to visual tasks. To understand the reasons may help researchers to design more robust algorithms which will eventually lead to an automatic visual system. This approach to artificial intelligence, and in particular computer vision , is not new. In the 1950s and 1960s, researchers using such methods tried to mimic human vision [91]. After a gap of about ten years, this line of research was revived in the early 80s. In this chapter, the research outcomes from the neurophysiology and psychophysics of vision will be examined and a class of artificial neural networks which have potential applications to visual tasks will be discussed.

3.2 Properties of the Human Visual System

Early models of the human brain led to a Von Neumann type computer, which contains a powerful CPU for processing data. While this type of computer is very successful in numerical computations, it is less effective in vision, which is so natural to human beings that its complexity tends to be underestimated. Although there was some success with the traditional approaches, a visual system as robust as the human visual system is still elusive, even after research of several decades since the sequential digital computer emerged. Re-examining the vision problem, researchers have agreed that vision in the HVS is the result of interaction of several adaptive visual processes which are highly parallel [111] [56]. Unfortunately, the solution to vision provided by evolution is far from being understood, even though a lot is known about the early stages of vision. Indeed, the human brain contains over a hundred types of neurons and in total, about 10^{12} nerve cells. This huge number of neurons and their complex connection networks yield a system which is fault-tolerant and highly parallel, but also difficult to analyze. However, with advances in neurophysiology, some details of biological networks have been revealed. Better algorithms are made possible by taking into account these properties found in the human visual system. With this in mind, it is worth having a brief review of the HVS.

3.2.1 Biological Neurons

The basic building blocks of a biological neural network are neurons. A neuron typically consists of nucleus, cell membrane, axon and dendrites (see figure 3.1). The axon is the channel through which the neuron relays its activity to other neurons. The activity is transferred in the form of a train of impulses[1]; in other words, it is frequency coded. There are gaps between the terminals of the axon and dendrites of other neurons. These gaps, called synapses, have been suggested to provide one of the mechanisms for adaptation [43]. Across a synapse, the activity of the *presynaptic* neuron is conveyed by means of chemical transmitters to the *postsynaptic* neuron. The chemical transmitters make the postsynaptic neuron either more or less likely to fire impulses. If the change is positive, the synapse is excitatory; otherwise, it is inhibitory.

In general, the activity[2] of a neuron is determined by the totality of excitatory and inhibitory signals exerted by other neurons through the synapses. When the stimulus to the neuron is great enough, the neuron in turn will fire impulses[3] to relay the information to other neurons. Figure 3.1 shows the relation between the firing frequency and the cell potential of a neuron. For more detail, see [96].

3.2.2 The Human Visual System

Figure 3.2 shows the schematic structure of the human visual system. Much less is known when it comes to the whole system. The following is a summary of the human visual pathway(for more details, see [53]) , in particular, those properties concerning edge detection and shape recognition [44].

[1] There are cells, for example, horizontal cells in the retina, which have no all-or-none action potential [96]

[2] The integrated potential

[3] the action potential is all or none .

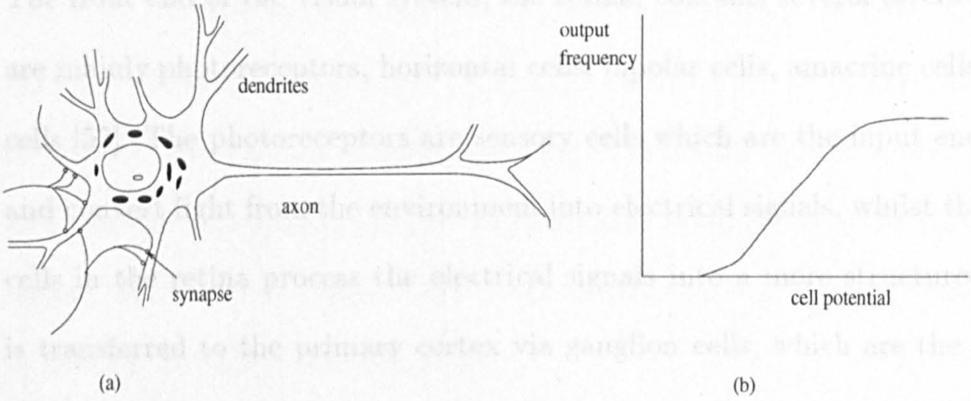


Figure 3.1: (a) A neuron (b) Its output frequency vs potential.

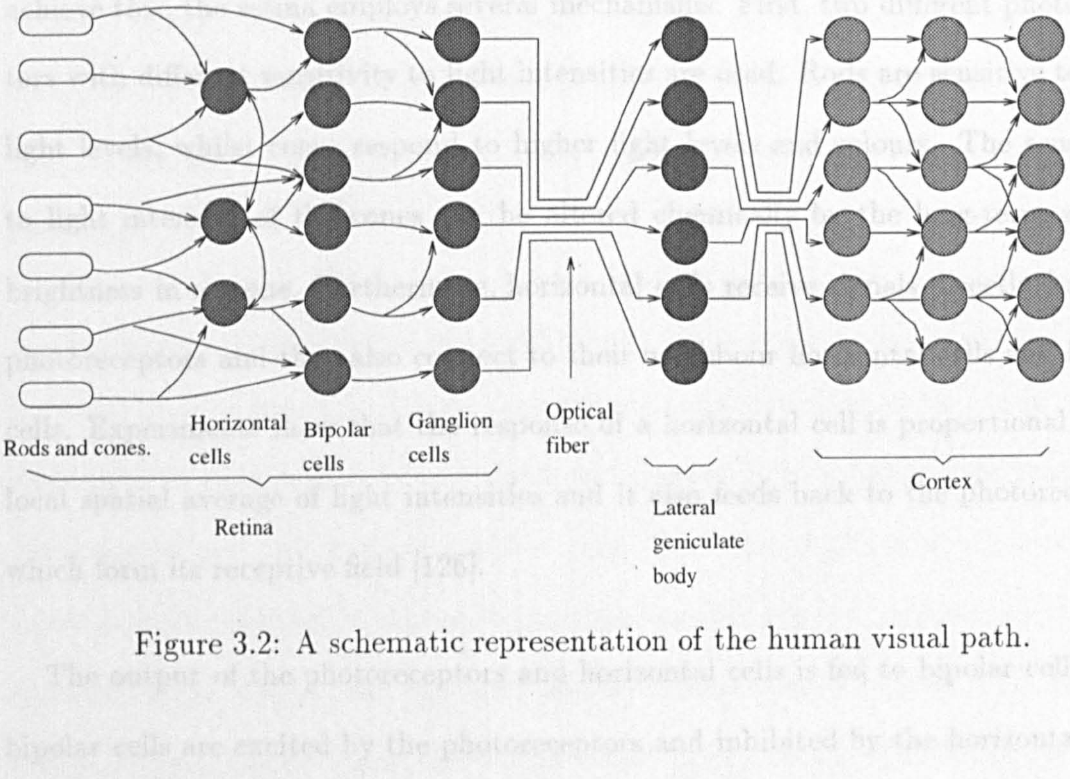


Figure 3.2: A schematic representation of the human visual path.

The Retina

The front end of the visual system, the retina, contains several layers of cells, which are mainly photoreceptors, horizontal cells, bipolar cells, amacrine cells and ganglion cells [53]. The photoreceptors are sensory cells which are the input end of the retina and convert light from the environment into electrical signals, whilst the rest of nerve cells in the retina process the electrical signals into a more structured form, which is transferred to the primary cortex via ganglion cells, which are the output end of the retina. Information flows in the retina both vertically (from one layer to the next layer) and horizontally (among neighbour cells in the same layer).

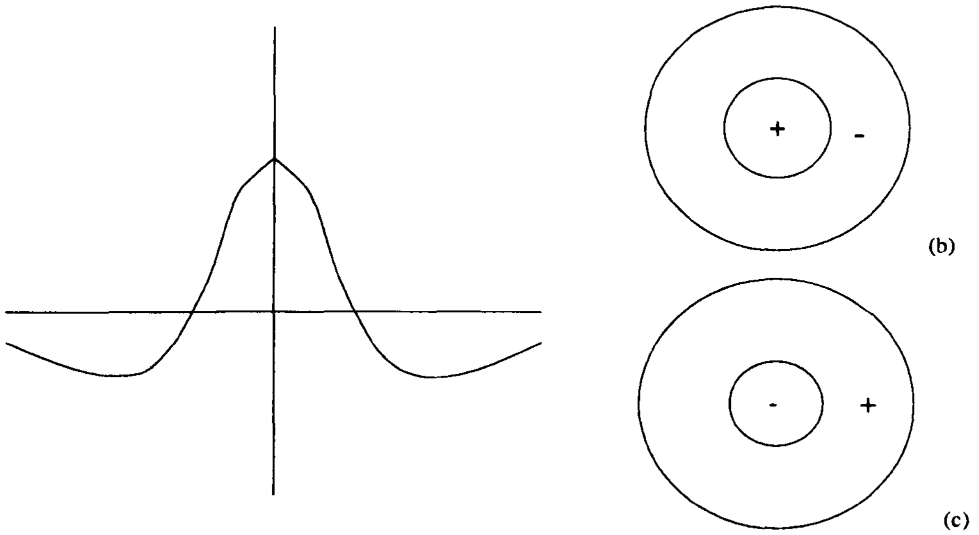
A crucial function of the retina is adaptation, which enables the brain to see objects in different lighting environments ranging from a starlit night to a bright noon. To achieve this, the retina employs several mechanisms. First, two different photoreceptors with different sensitivity to light intensities are used. Rods are sensitive to lower light levels, whilst cones respond to higher light levels and colours. The sensitivity to light intensity of the cones can be altered chemically by the long-term average brightness in a scene. Furthermore, horizontal cells receive signals directly from the photoreceptors and they also connect to their neighbour horizontal cells and bipolar cells. Experiments show that the response of a horizontal cell is proportional to the local spatial average of light intensities and it also feeds back to the photoreceptors which form its receptive field [126].

The output of the photoreceptors and horizontal cells is fed to bipolar cells. The bipolar cells are excited by the photoreceptors and inhibited by the horizontal cells.

Because the photoreceptor and the horizontal cells have logarithmic responses to light intensity, the output of a bipolar cell is the ratio of local light intensity to background intensity. In other words, bipolar cells respond to normalized light intensities rather than the absolute light intensity. The implication of this local adaptation is that the retina not only ensures reliable signalling of small changes in image brightness but also enhances features in images. Knowing these properties of the three layers of the retina, Mahowald and Mead built a silicon adaptive retina whose behavior is similar to that of the biological system [85].

The output end of the retina is the ganglion cells, whose axons form the optical fibres heading to the lateral genicular body. There are no photoreceptors at the place where the optical fibres come to form a bundle and head to the primary visual cortex. Psychophysiological experiments show that when an object falls into that spot, it cannot be seen. However, an object such as a line which runs across the spot will not appear broken. In other words, there may be a mechanism in the higher levels of vision which fills the gap. It also seems to imply that the visual system uses *a priori* knowledge [25].

The distinct characteristics of ganglion cells are that they respond to moving objects and their receptive fields have a center-surround structure, so that they respond maximally to change of light intensities and minimally to uniform light intensities. There are at least two types of ganglion cells. One type of ganglion cell responds to a center-on-surround-off receptive field, while another type of ganglion cell has the opposite receptive field, that is, center-off-and-surround-on (See figure 3.3). Study of optical illusions suggests that the human retina plays the role of reducing the band-



(a) The cross-section of the recptive field of a ganglion cell

Figure 3.3: A schematic representation of ganglion cells' receptive field

width of visual data and extracting only these essential features of an image. Illusions are created because the retina selectively encodes information. Marr thus proposed that a $2\frac{1}{2}$-D primal sketch is used by the human visual system [88]. Marr suggested that these center-on-surround-off cells together can detect zero-crossings, which correspond to edge locations [88]. Furthermore, Marr suggested a 'raw primal sketch' built upon these zero-crossing locations. The center-on-surround-off characteristics were also suggested as the mechanism for orientation preferences of simple cells found in the visual cortex [53] (see figure 3.5).

The Primary Visual Cortex

In the previous subsection, an account was given of the functional structures of the retina. The optical fibres convey the encoded information to the next stage of the visual pathway, the lateral geniculate body (LGN). The lateral geniculate nuclei fur-

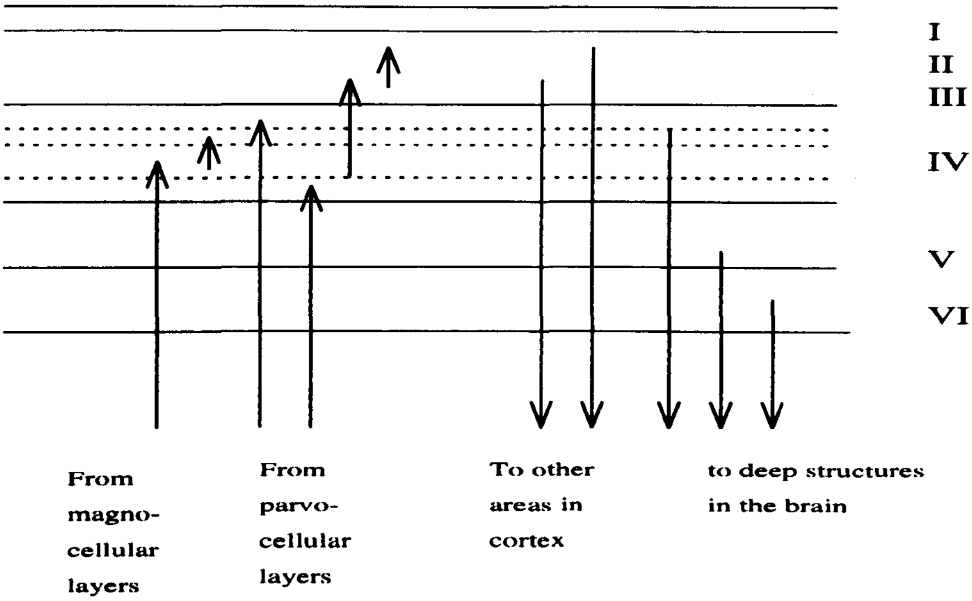

I
II
III
IV
V
VI
From magno-cellular layers
From parvo-cellular layers
To other areas in cortex
to deep structures in the brain

Figure 3.4: Cortex layers

ther transmit information to the primary visual cortex(the striate cortex) and they also receive feedback from the primary visual cortex. Very little about the functions of this body is known. Further into the visual cortex, structures become complex and the degree of abstraction also increases. The neurons start to show highly task-specific characteristics. Two major transformations of information are accomplished in the visual cortex. The first one is the rearrangement of incoming information, so that most of its cells respond not to spots of light, but to specifically oriented line or edge segments. There are cells which are orientation specific and their complexity varies. Anatomically, the visual cortex consists of 6 layers of cells(see figure 3.4). The axons from the geniculate body terminate at the layer IV. Cells in this layer have center-surround receptive field structures like those of ganglion cells. A layer further, cells possess different receptive fields and their complexity also increases. Cells with simple orientation-specific receptive fields are called simple cells[55]. It is suggested that simple cells behave as if they received their input directly from several cells with

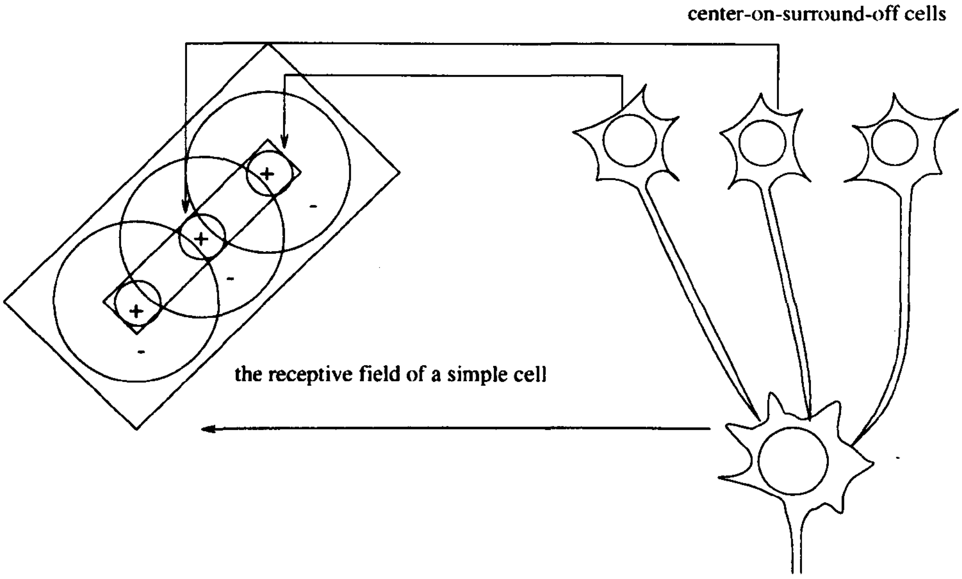

center-on-surround-off cells
+
+
+
-
-
-
the receptive field of a simple cell

Figure 3.5: A suggested simple cell receptive field

center-surround circularly symmetrical fields, which are found in layer IV [53]. The suggested scheme is illustrated in figure 3.5. Cells in the next stage are called complex cells, which share with the simple cells the quality of responding only to specifically oriented lines. However, instead of responding to an appropriately oriented stationary line, complex cells respond to an appropriately oriented moving line independently of its position within their receptive field. Some complex cells also show directional selectivity, i.e. they only respond to oriented lines moving in one direction but not in the opposite direction. Unfortunately, how these orientation-sensitive cells are organized to produce boundary representations of the world is unknown.

Among different models, Hoffman proposed the Lie transformation group model of neuropsychology to represent and explain the locally smooth processes observed in the visual field and their integration to the global field of visual phenomena [47] [30]. In chapter 4, integration of local boundary segments in an image is used for edge detection.

Another important feature found in the striate cortex is the presence of end-stopped cells. Cells with end-stopping properties are suggested to relate with end-point detection. Dobbins *et al.* proposed such cells are used for coarse curvature estimates [29].

The visual cortex contains systematic 2-D maps of the world which it represents. This follows from the fact that neighbouring neuron cells in this area have neighbouring receptive fields. This topographical mapping, or retinotopic map, has attracted much attention[69]. Figure 3.6 is the schematic representation of an example of such a topographically organized map. Cells responding to similarly oriented lines are neighbours. Also, this schematic representation of the primary visual cortex shows the periodicity of ocular dominance: the combination of information from two eyes to give depth perception is not a concern of the present work, however. Readers who are interested in this can find details in references [53] [63].

Beyond Striate Cortex

The cells in the striate cortex further project into the prestriate cortex and other parts of the brain. How the cortex integrates the low level information to represent its surrounding world and recognize objects is currently an area under intensive study. By examining a few visual cortex-damaged patients, it has been suggested that the visual cortex is divided into several task-specific parts and that each part contributes to visual perception [138]. Damage of a special part of visual cortex will cause the

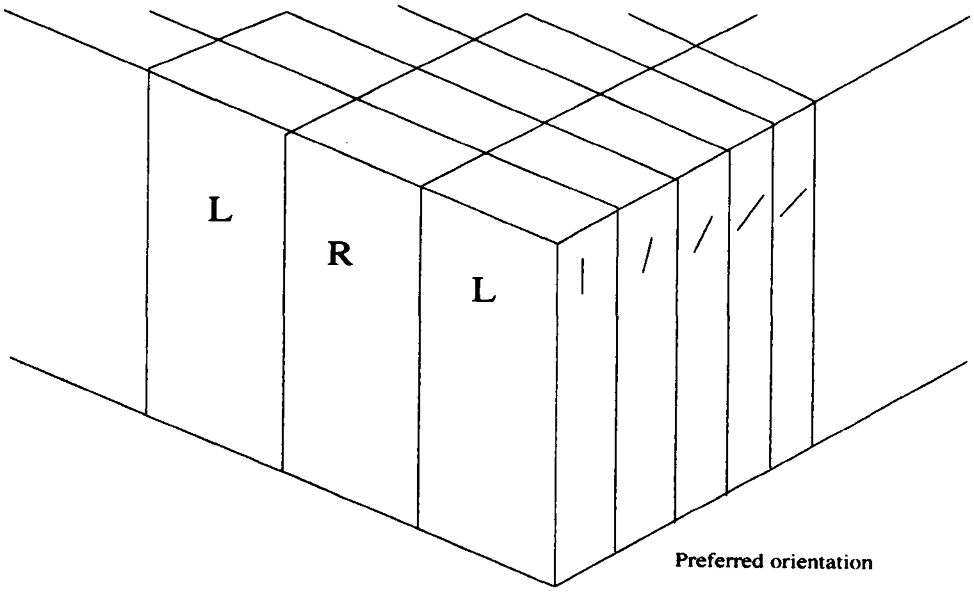


Figure 3.6: The cortex map; Left and right ocular dominate and orientation specific columns

patient to suffer certain disabilities of vision and perception[138]. The study also suggests that the higher level processes have some influence on the low level processes.

3.2.3 Summary

A distinct property of the visual pathway is that cells in higher levels appear to 'see' a greater perspective than those at an earlier level and have increased ability to abstract [63] [53]. This suggests a hierarchical structure for artificial neural networks. At all stages of the visual pathway (retina to cortex), there are cells with different sizes of receptive fields [53]. This fact is used to support the use of multiresolution representations and models for image processing and computer vision.

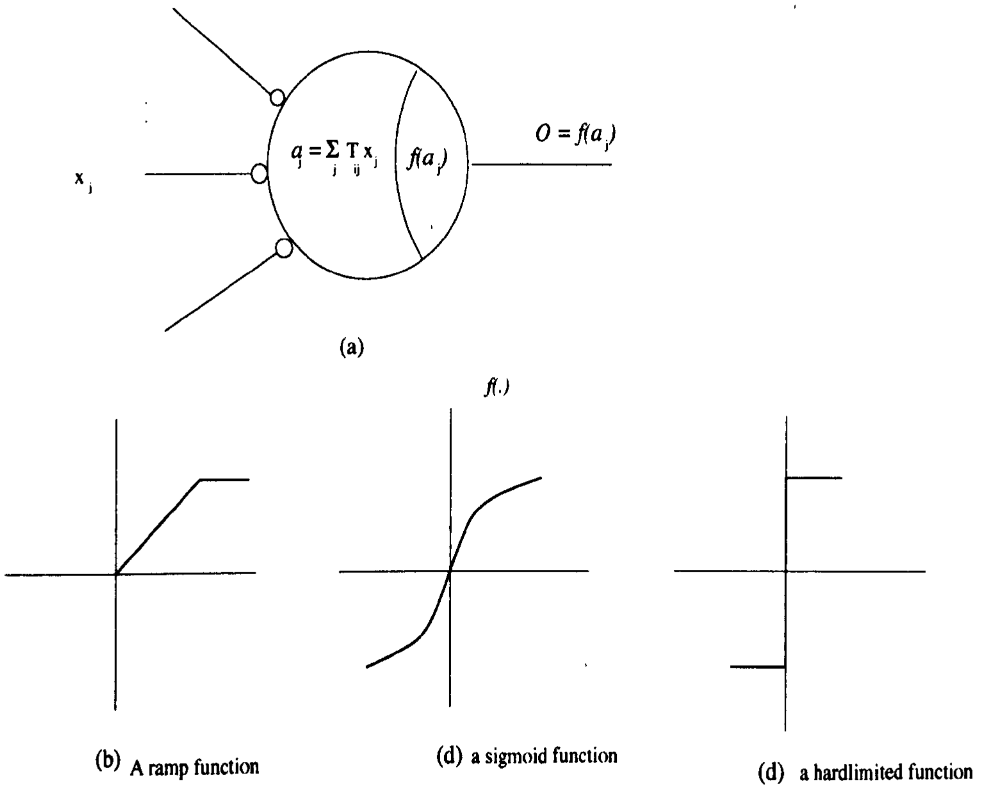
It is arguable that a biological neural network is the only efficient solution to the vision problem. However, it seems reasonable to borrow these techniques invented by evolution. Thus, the question is what are the essential characteristics that make a biological network better than an traditional computer algorithm? Apparently, the learning and adaptation abilities of biological neural networks are responsible for their robust performance in a changing environment. Psychological studies show that cultural differences influence human visual perception [44]. This seems to support the idea that learning plays an important role in perception and vision.

Consequently, it would be helpful to know what is the mechanism for adaptation and plasticity or learning. The mechanism for learning is unfortunately, little known, although suggestions that synapses are the candidates for learning have been made [53]. For adaptation, it seems that competition is responsible for adaptation and lateral inhibition is naturally the device which enables competition among neurons.

3.3 Modelling Biological Neural Networks

3.3.1 The Computational Neuron

The most realistic model should model every observed biophysical behaviour from real neurons. However, the details of electro-chemical processes in a neuron are very complicated and there are over a hundred different types of neurons. A detailed model may not be useful in better understanding the overall network behaviour. Indeed, the detailed model may have too many parameters, making analysis difficult, if not impossible. Thus, simpler versions of the neuron model are widely used in the artificial neural network community. Figure 3.7 shows a McCulloch-Pitts neural model, which is a typical neuron model. Also shown are typical limiting functions.



(b) A ramp function

(d) a sigmoid function

(d) a hardlimited function

Figure 3.7: (a)McCulloch-Pitts neuron model. (b), (c) and (d) nonlinear output functions.

Generally, an artificial neuron is a first approximation of a real neuron and is a simple computational unit, which responds to stimuli in the following fashion: a neuron's activity is the sum of a bias and a set inputs which are the products of the connection weights and signals channelled in through the respective connections. Then, the output of the neuron is a function of its activity. Usually the mapping will be a nonlinear one (see figure 3.7). The neuron's activity y_i is described by the following linear differential equation,

$$\frac{dy_i}{dt} = -\eta y_i + [\sum_j x_j w_{ij} + \theta_i] \tag{3.1}$$

where x_j is the input signal channelled in through weight w_{ij}, θ_i is a bias value and η a scale factor. Several versions of this equation can be found in [39]. Further simplifying the neuron's activity model by dropping the decay term $-y_i$ gives the commonly used neuron activity model, McCulloch-Pitts model, used in computer simulations,

$$y_i = \sum_j x_j w_{ij} + \theta_i \tag{3.2}$$

An important generalization of the additive model (equation (3.1)) is the shunting model, which mirrors the underlying physiology of single nerve cell dynamics (see Hodgkin-Huxley [46]).

As noted, the neuron is a simple computational unit and equation (3.2) can be viewed as a match operation between input signal vector X and the weight vector W

as follows

$$y_i = \sum_j x_j w_{ij} = X \cdot W_i = \|X\| \|W_i\| cos(X, W_i) \tag{3.3}$$

Equation (3.3) can also be considered as a linear filter operation in image processing. Hence, a network of such neurons with their connection weights set properly will be able to perform many image processing tasks. For example, combined with competitive training, Kohonen proposed a network for adaptive coding [69].

While the McCulloch-Pitts model has been applied to many neural computing systems successfully, it is an over-simplified version of a true neuron. Many aspects of a biological neuron remain to be explored. For example, time dependent properties of biological neurons are common and are likely to be important to their computational abilities.

3.3.2 Learning and Adaptation

How a biological neural network learns has not been exactly pinned down yet, but it is a common opinion now that synapses are one of the learning mechanisms in the neural networks. Hebb in his classical book [43] –"The Organization of Behaviour", gave his conjecture of neural learning:

> *When the axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it , some growth process or metabolic change take places place in one or both cells such that A 's efficiency, as one of the cells firing B, is increased.*

Hebb also suggested that changes in the efficacy of synapses could take place via growth of synaptic knobs. However, such growth has not been commonly observed to happen in adult animals. Recent discoveries suggest that different rules are used to modify the strength and patterns of connections between neurons [117][62]. Thus, many alternative mechanisms have been proposed for changing synaptic efficacy. And there has been just as wide a variety of mathematical rules proposed for such modification. Nevertheless, Hebb's conjecture of neural learning has been widely adopted to train artificial neural networks, including Hopfield networks. As learning may be much influenced by higher level cognitive processes, the reader can see [90] for more details.

3.4 Hopfield Neural Networks

As mentioned in section 1.5, a Hopfield neural network is a single layer recurrent network (see figure 3.8). Originally, the Hopfield neural network is related to the Ising-spin model used for describing the binary spins of magnetic atoms and was proposed as a form of associative address memory [48]. This type of network is useful in pattern recognition (eg. OCR), although it has been proved that the network's storage capacity is as low as about 15% of its total number of units [92]. Later, Hopfield and Tank proposed using the Hopfield network to solve combinatorial problems [50]. Such an approach relies on defining proper connection weights and is adapted to solve image processing problems which can be cast as quadratic optimization problems [141] [134] [58].

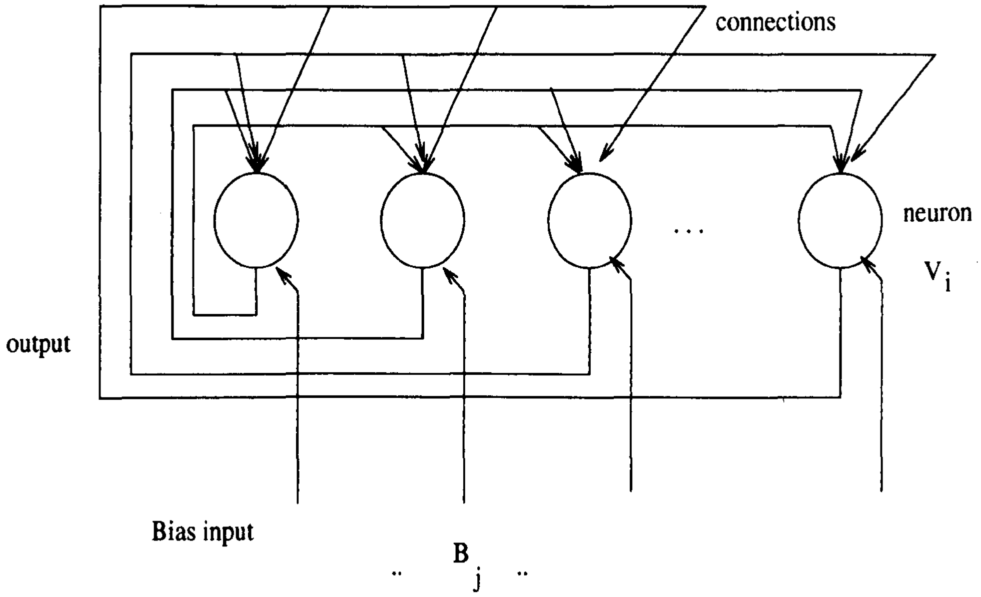


Figure 3.8: The Hopfield neural network structure

A Hopfield net of n nodes is a fully connected network which can be represented by a graph $G = (V, \mathbf{T})$ where $V = \{V_1 ... V_n\}$ is a n-tuple vector whose element V_i represents the output of node i, and $\mathbf{T} = \{T_{11} ... T_{ij} ... T_{nn}\}$ is a $n \times n$ symmetric matrix whose element T_{ij} represents the link weight between node i and node j. The output V of these nodes is a function of their potentials U. This function can be continuous or binary.

The Continuous Model

The continuous Hopfield network was proposed for VLSI implementation [49]. It consists of nodes whose potentials are described by

$$C_i \frac{dU_i}{dt} = -\eta U_i + \sum_j T_{ij} V_j + B_i \qquad (3.4)$$

where B_i is the bias input, and C_i and η are parameters. The outputs of these nodes are

$$V_i = f(U_i) \tag{3.5}$$

where $f(\cdot)$ is a differentiable function, for example a sigmoid function

$$f(U_i) = \frac{1}{1 + e^{-U_i/T}} \tag{3.6}$$

For the evolution of the network, Hopfield defined an energy function E which is a Lyapunov function when T_{ij} is symmetric[49]. It is as follows:

$$E = -\frac{1}{2}\sum_{i,j} T_{ij}V_iV_j - \sum_i V_iB_i + \eta\sum_i \int_0^{V_i} f^{-1}(V_i)dV_i \tag{3.7}$$

However, the term $\sum_i \int_0^{V_i} g^{-1}(V_i)dV_i$ is usually neglected by setting $\eta = 0$.

The Discrete Model

In a discrete Hopfield net, the output function $f(\cdot)$ is a step function

$$f(U_i) = \begin{cases} 1 & \text{if } U_i \geq 0.5 \\ 0 & \text{if } U_i < 0.5 \end{cases} \tag{3.8}$$

In addition, U_i is updated in discrete time steps,

$$U_i(t+1) = \sum_{j=1}^{n} T_{ij}V_j(t) + B_i \tag{3.9}$$

that is, the potential of node i at time step $t+1$ is the linear weighted sum of elements of V at time step t. The energy function for a discrete model is

$$E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} T_{ij}V_iV_j - \sum_{i} B_iV_i \tag{3.10}$$

The dynamic evolution of the Hopfield neural networks is in the direction of reducing the energy function. However, the convergence properties and stability of Hopfield networks depend on the update mode and the structure of the matrix T of connection weights [13]. For the continuous model, it is possible to use asynchronous or synchronous update. For the discrete model, Hopfield proposed an asynchronous update scheme, which is summarised as follows:

1. Randomly choose a node.
2. The activity of the chosen node is computed using equations 3.9 and 3.8.
3. If the total energy of the system decreases then the neuron's state is changed; otherwise, its state is not changed.
4. Repeat 1-3 until there is no more change.

The above update scheme is a serial model. The convergence property of updating in a parallel or synchronous model for the discrete Hopfield model was also investigated by Bruck [13] and Paik [97].

From equation (3.9) and (3.10), it can be seen that a Hopfield net performs a kind of gradient-descent search in its energy landscape, which only gives a local minimum of the energy function. A Hopfield network using simulated annealing methods for dynamic evolution, also known as a Boltzmann machine, may achieve a better performance, but with a price in computation time [1]. Nevertheless one of the main attractions of the Hopfield network for low level vision is that it uses 'neurons' in a well defined way, to minimize an energy function. It therefore has the potential to exploit the neural metaphor of computation in a way which is well founded mathematically.

3.5 Regularization and Hierarchical Neural Networks for Multiresolution Models

3.5.1 Regularization Theory

As discussed in Chapter 1, image processing systems, whether biological or artificial, are often used to extract information from available image data. Because of noise and the loss of information in the imaging process, inverse processes which try to recover the information from 2-D images are ill-posed, i.e. there is no unique solution [104]. This ill-posed problem is often formulated as follows [104]:

Given data Y and a transformation $\mathbf{A}$ find X such that $\mathbf{A}X = Y$.

A direct inverse matrix computation $X = \mathbf{A}^{-1}Y$ is often impractical when the matrix is large. In ill-posed cases, when A is singular, only an approximate solution, which minimizes the following quadrature cost function E is usually sought,

$$E(X) = ||\mathbf{A}X - Y||^2 \tag{3.11}$$

where $\| \cdot \|$ is a norm. Furthermore, when A is ill-conditioned, the solution from a direct inverse matrix computation is usually unstable, i.e. a small change in data Y will make a huge change in the solution. This is to say that the method is not robust. In order to make an ill-posed problem well-posed, the admissible solutions must be restricted by introducing suitable constraints, or *a priori* knowledge. *A priori knowledge* can be given, for example, in the form of variational principles or statistical properties of the solution space. Recently, Poggio proposed regularization theory as a theoretical framework to unify work in various early vision processes [104]. In regularization theory, one technique is to integrate image data and various constraints using the framework of minimizing an 'energy' or cost function [64]. In such approaches, the relation between each clue or constraint is expressed by a cost function in such a way that compatible constraints and clues will reduce the energy function while incompatible ones will increase the energy function. The solution will be the one which minimizes the energy function.

One way to restrict the possible solution space is to find X that minimizes

$$E(X) = \| \mathbf{A}X - Y \|^2 + \lambda \| \mathbf{P}X \|^2 \tag{3.12}$$

where $\mathbf{P}X$ is a regularization term, and λ is a so-called regularization parameter which controls the trade off between data fidelity and constraints. A large λ means that the solution is forced to pay more attention to the constraints than to the data, Y.

If **A** is a linear operator, the norm quadratic and P linear, the paradigm is called a standard regularization by Poggio *et al.* [104]. Standard regularization under some mild conditions can be shown to have a convex solution space and, therefore, a unique solution exists. However, the physical plausibility of the solution, rather than its uniqueness, is the most important concern in regularization analysis. Naturally, non-quadratic cost functions may be needed to impose correct physical constraints. However, the solution space is then no longer convex and there may be many local minima.

This energy minimization paradigm is general and its effectiveness in a particular problem depends on the formulation of the cost functions corresponding to each clue. Because images usually consist of textures or structured objects, pixels in a neighbourhood tend to have high correlation. Thus it seems reasonable to construct an energy function corresponding to such properties. Consequently, a natural choice among a variety of minimization techniques is that which best reflects such local interactions among pixels. To implement the minimization, it is common to use iterative methods like relaxation, simulated annealing and optimization networks [22]. Certainly, these techniques are related, but with some differences and they have all been successfully applied to various problems (eg. [143] [35] [50]). As suggested in previous sections, the use of artificial neural networks may be an efficient way to solve image processing problems. There are also theories that suggest the brain behaves to optimize a single variable, even though such claims have been challenged [77]. The utility of neural networks in optimization is still an appealing idea. Since the energy function of a Hopfield network and a standard regularization equation are both quadratic, Poggio and colleagues[104] suggested that the Hopfield neural network can be used to solve standard regularization problems after a careful mapping between the Hopfield network and the standard regularization. It is one of the best understood parallel methods to implement regularization and hardware implementations are available [58]. There are many researchers using Hopfield networks for image processing. For example, Zhou *et al.* [141] applied a Hopfield network to an image restoration problem. Psarrou and Buxton combined Geman and Geman's line process with a Hopfield net for optical flow estimation [105] [35].

3.5.2 Hierarchical Hopfield Networks

From chapter 2 and the previous part of this chapter, it has been suggested that combining local and global information is important both for a computer program and a biological system in image processing and vision. Since both a multiresolution model and a layered neural network can be represented by hierarchical structures, a layered neural network is easily adapted to implement a multiresolution model. A simple way to implement a multiresolution model using neural networks is to create a neural network for each resolution. Among the various artificial neural networks, the Hopfield network is easily extended to be hierarchical and embody a multiresolution model. Such an approach has been used by a few researchers(eg. [134] [31] [7] [5] [27]). Wu formulated an optical flow reconstruction as a optimization problem based on a regularization theory and employed a hierarchical Hopfield neural network to find the solution[134]. While using the same principles, however, Battiti employed a multilayer perceptron and used a multiresolution pyramid for computational efficiency in optical flow reconstruction[7]. The benefit of such a combination is to increase the convergence rate of the neural network as well as to enhance the computation power of the multiresolution model through local interactions.

A hierarchical Hopfield neural network which implements a multiresolution model to solve image tasks, such as image restoration, usually needs self-looped connections. This implies that an asynchronous network would not be guaranteed to converge without checking the change of its energy function [141]. Several researchers have studied algorithms to eliminate this problem [97] [136]. It is reported that a binary Hopfield network will converge to a fixed point provided that its connection weights matrix is symmetric with non-negative diagonal entries and it operates in a sequential mode[14]. Paik and Katsaggelos also investigated the convergence properties of a binary Hopfield network using different updating rules when applied to image restoration problems [97]. Their result is an improvement and extension of that of Zhou et al.

Although the connection matrices of a Hopfield network often contain negative diagonal entries when using standard regularization techniques to solve an image restoration problem, this is not a severe problem. Figure 3.9 illustrates the aforementioned hierarchical Hopfield neural network, which corresponds to a pyramid. Finally, the multiresolution Hopfield network may have the benefit of fast computation. A multiresolution neural network architecture with coarse-to-fine computation can improve the convergence rate of the Hopfield network[93]. For all of these reasons, the network chosen for the work reported here employs the structure shown in figure 3.9.

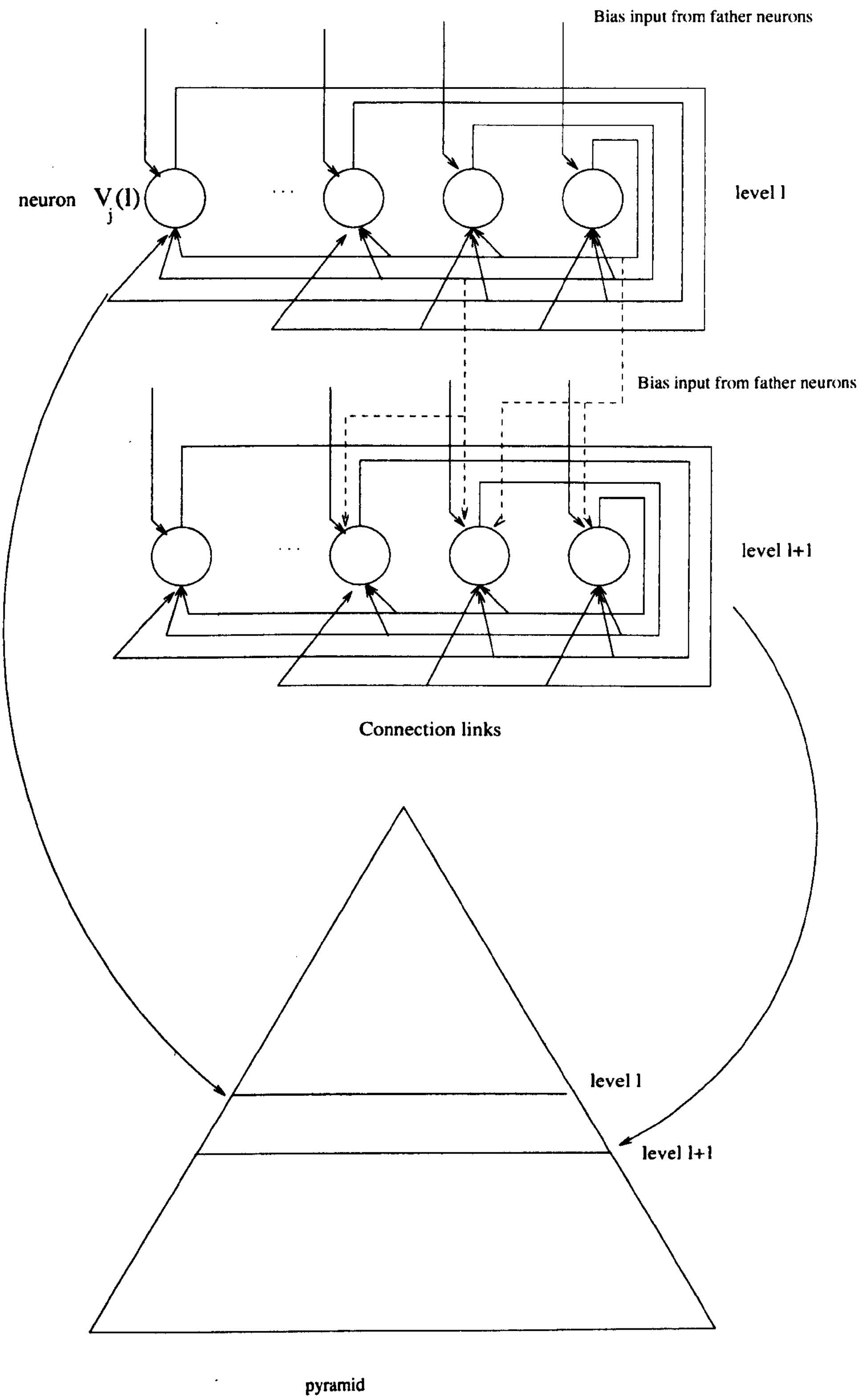


Figure 3.9: A hierarchical Hopfield neural network

Chapter 4

Multiresolution Edge Detection Networks

4.1 Introduction

As mentioned earlier, boundaries, along with some other properties, are used by a visual system to represent its environment. The task of boundary extraction is to transform an intensity image into a binary labelled edge map of the image. Edge detection is the first step of the boundary feature extraction process. Previous work on edge detection ranges from the earliest gradient operators [61] to more recent model-based methods using relaxation [41] [142] or multiresolution representations [18] [87]. In this chapter, it will be shown that gray level edge detection can be done well with a biologically inspired approach. This approach is based on the framework of minimization of an energy function which corresponds to edge configurations in the image. Using a multiresolution model to enhance orientation information, the minimization is implemented by mapping the cost function into the energy function of the hierarchical Hopfield neural network proposed in chapter 3.

4.2 Edge Detection

The aim of edge detection is to register possible boundaries of objects in a given image for higher level visual processes to represent objects. Since the terms, 'boundary' and 'edge' are often used interchangeably, it is not surprising that there is not yet an generally agreed definition of 'edge'. However, it is hard to disagree that an edge is, roughly speaking, a location where image properties such as intensity, colour, or texture, change abruptly.

Although simple and intuitive, this definition is fuzzy. Further examination of the definition and comparing edge drawings of images by humans, it is not hard to see that an edge map produced by strictly following the above edge definition can be very different from edges perceived by a human observer. It is thus not surprising that giving an image to two individuals, they may produce two similar but not identical edge drawings. This is because individuals may draw boundaries which are the contours of objects perceived in higher level visual processes, but not those fine details such as texture and shadows, which also mark abrupt changes of image properties. The point here is that edge detection is the first step of boundary extraction. Hence, it is reasonable to postulate that good edge detection should give an edge drawing as close as possible to the boundaries of objects in an image and to incorporate some *a priori* knowledge to enhance the performance of an edge detection algorithm.

The above definition implies that edges occur at the locations of large intensity gradient. This observation naturally leads to the earliest gradient schemes [61] and later a class of optimal filtering edge detectors [38][67][20]. However, the inherently differential nature of edge detection makes it very sensitive to noise. As Poggio pointed out [104], the edge detection problem is ill-posed, in the sense that a small amount of noise will cause a large change in the resultant edge map. Indeed, noise will cause large gradients, which disrupt the small scale edges. To be insensitive to noise, a straightforward approach is to apply lowpass filters to the image first before applying a differentiation operator. Unfortunately, such operations will also increase the uncertainty of the location of edges [38] [20] [131]. Thus, difficulties arise in optimal filtering detectors, because of the compound problem that edges which are perceived and found meaningful often exist in a range of scales [88] [132] and images are often blurred and corrupted with noise. To tackle noise sensitivity and maintain the accuracy of edge locations, multiresolution techniques have been employed [110][132][127]. Witkin proposed the scale-space filtering scheme to cope with the so called scale consistency problem and alleviate noise sensitivity. The *scale consistency* problem, however, has to be tackled efficiently in order to produce an unambiguous result. To this end, various methods for the combination of information between scales have recently been proposed. Gidas used the RG method to generate a multiscale structure and related the scales to one another via the RG transformation [37] Another approach is to model scale space using a stochastic process. Notable are the work of Clippingdale and Wilson [24] on a quadtree structure and that of Basseville *et al.* [6] on a wavelet representation (cf. chapter 2). Using a pyramidal representation and a stochastic multiresolution model which is based on Clippingdale's work, Bhalerao [10] proposed a scheme which achieved some degree of success in extracting edges.

Edge maps obtained by applying derivative operators are followed by a grouping process which takes orientation information into account to give a set of edge chains [66]. For example, a classical clustering method, the Hough transform, [59] maps potential edge pixels of a local region into a parameter space of curves, and an edge chain is formed by selecting the maximum peak in the parameter space and represented by piecewise analytic curves such as straight or spline lines. The Hough transform, however, requires a search for peaks, which is time consuming, especially if the resolution required is high.

Another classical approach of grouping edge pixels is to cast the edge grouping problem as an optimization problem. The idea is to formulate a cost function of an edge structure such as, local position, orientation and curvature, and to find a solution by using minimization techniques such as the relaxation labelling used Parent and Zucker [102], and Haddon and Boyce [41], dynamic programming [3] or simulated annealing [119].

Obviously, the major issue in edge grouping is to *regularize* the ill-posed edge detection problem. In other words, knowledge of edge structure is used in order to achieve noise immunity. Therefore, the relevant properties of edges warrant further investigation.

While noise and fine texture usually result in random short line segments, most physical boundaries of interesting regions are more or less smooth and continuous. It is thus desirable that edges should also be smooth and continuous. Such a characteristic can be used to distinguish an edge from short line segments caused by fine texture and noise, by discarding incoherent or isolated short line segments, or connecting neighbouring short line segments which have similar orientations. This is a widely used criterion for edge grouping. For example, Boldt et al. [12] considered short lines as tokens and used geometrical relations for line grouping.

Although the magnitude of a gradient is strongly influenced by illumination, its orientation is relatively independent of the illumination and provides shape information. With the discoveries of simple cells [55] which respond to linear features whose orientations lie within a narrow angular band, it is not surprising to see that this essential parameter of edges is important in edge detection. Furthermore, psychologists have suggested that the human visual system seems to use the Gestalt laws in partitioning elements into groups; these include proximity, similarity, closedness and good continuity [88]. In short, in addition to magnitude, there are four essential properties which can be used to discriminate edges from noise. They are (a) orientation, (b) thinness, (c) continuity and (d) length. These properties should be included in the design of an edge detection algorithm.

4.3 Edge Detection as an Optimization Problem

As mentioned earlier, edges can be defined as the locations of boundaries across which one or more image attributes are discontinuous. It is nevertheless difficult to quantify what is an edge in an image since the perception of edges by the HVS is very complex and is strongly influenced by prior knowledge [88]. A simple way to find edges in an image is to filter the image and threshold the result. However appealing its simplicity, this approach fails to take structural properties into account and is not robust when noise and fine texture is present [61] [132]. This leads to a range of algorithms which further group edge pixels into connected lines using curve fitting, sequential contour tracing techniques or cost minimization techniques, in which the edge discrimination problem is cast as an optimization problem.

The idea of formulating the edge detection problem as an optimization problem is appealing, for a large family of optimization techniques can then be utilised. The main theme is to formulate an energy or cost function which will reflect the characteristics of an edge or a boundary. For example, Parent and Zucker use the concept of *cocircularity* to define interactions between edge pixels and the sum of these interactions is used as the global energy function which is minimized, using a relaxation labelling method, to give the edge configuration [102]. Using the same idea, Amini *et al.* applied dynamic programming techniques to solve the problem. Also using dynamic programming for curve inference, Shaáshua and Ullman [116] proposed a uniform network of locally connected processing elements to compute the structural saliency in an image. Later, Montesinos and Fabre [94] modified the technique used by Shaáshua and Ullman [116] also for grouping edge elements which, unlike the line segments used by Shaáshua and Ullma, included dots. In practice, these optimization techniques may only give local minima and the goodness of the solution provided depends on the initial guess.

The essential task in casting an edge detection problem as an optimization problem is to derive a cost function which reflects the edge structures and is easy to compute. It is also desired that the energy function should be easily embodied in a multiresolution model so that a good initial guess can be made at a low resolution and a better solution can be obtained by refining it at smaller scales.

4.3.1 The Design of a Combinatorial Energy Function

A great deal of work has been done since Hopfield neural networks were applied to optimization problems [50][141][137]. The attraction of using a Hopfield neural network is that optimization problems can be solved in a parallel manner and hardware implementation is readily available [49]. In image processing, many problems can be cast as optimization problems and Hopfield neural networks are used to find solutions. It is convenient to map pixels to nodes one-to-one, so that the interaction and constraints between pixels are easily implemented by the strength of their links and some *a priori* constraints can be also implemented as bias terms.

For the sake of convenience, the energy function of the Hopfield network is rewritten here (cf. equation (3.10))

$$E = -\frac{1}{2}\sum_i \sum_j T_{ij} V_i V_j - \sum_i B_i V_i \tag{4.1}$$

$$V_i = f(\sum_j T_{ij} V_j + B_i) \tag{4.2}$$

Obviously, if orientation information, to which the simple and complex cells respond, is used to decide the strength of interaction, T_{ij}, it should give the sum $\sum_j T_{ij} V_i$ a meaningful result. Indeed, this is a logical way to counter noise by a simple smoothing operation. However, to facilitate a smoothing operation taking orientation information into account, the orientation representation should be implemented carefully, so that the average can be used as a 'certainty' measure of the presence of an edge pixel. To this end, an interesting orientation representation known as the *double angle* representation, first proposed by Granlund[38], is adopted in this work. The reason for using this representation is twofold: (a) to avoid ambiguity and (b) for efficient smoothing.

There are a variety of orientation estimation and subsequent smoothing methods in the literature. This is due to the fact that a variety of orientation representations were adopted. In the early days, the orientation vector field of an image $f(x,y)$ was often represented by the gradient field $(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y})$ rotated by $\frac{\pi}{2}$ [61]. Although this representation is simple, there is an intrinsic ambiguity in this representation scheme since the same orientation can be represented by a vector with either positive or negative sign. To filter out noise, a simple smoothing operation is to average the orientation field within an isotropic weighting window. However, the two problems together will not allow the smoothing operation to produce a meaningful average [128].

One remedy for these problems is to represent orientation by doubling the angle of each gradient vector. Thus, given an image $f(x,y)$ and a pair of orthogonal masks k_0 , k_1, the gradient field (g_0, g_1) is given by

$$g_0(x,y) = f(x,y) * k_0(x,y) \tag{4.3}$$

$$g_1(x,y) = f(x,y) * k_1(x,y) \tag{4.4}$$

where $*$ is the convolution operation and $f(x,y)$ is the intensity image. The double angled orientation vector $\vec{d}(x,y)$ of a pixel (x,y) may be computed as follows: Let

$$g(x,y) = \begin{pmatrix} g_0^2(x,y) - g_1^2(x,y) \\ 2g_0(x,y)g_1(x,y) \end{pmatrix} \tag{4.5}$$

which is equivalent to

$$g(x,y) = r^2(x,y) \begin{pmatrix} cos 2\theta \\ sin 2\theta \end{pmatrix} \tag{4.6}$$

where $r(x,y)$ is the magnitude of vector $(g_0(x,y), g_1(x,y))$ [10]. Then,

$$\vec{d}(x,y) = r(x,y) \begin{pmatrix} cos 2\theta \\ sin 2\theta \end{pmatrix} \tag{4.7}$$

In this representation, there is no sign ambiguity, so smoothing can be performed easily. It is shown by Knutsson [67] that this representation is equivalent to the use of a tensor representation. Note the orientation vector field is the continuous one, as opposed to a discrete one advocated by Zucker [143]. The advantage of using a continuous representation is to allow a simple local smoothing and avoid quantization errors. From the neuropsychological viewpoint, it is also suggested by Hoffman [47], and later Dodwell [30] that the integration of local orientation information in the visual cortex is to consider the cortex as a vector manifold (ie. a continuous vector field), on which vector fields operate, and to model the integration process as a function of the postulated vector fields and their global properties.

Compatibility Measurement

The properties of edges outlined in section 4.2 are used to define a measure of compatibility T_{ij} for two pixels in a window. If a Hopfield neural network is used, those structural properties have to be imposed as constraints as follows:

$$V_i = f(\sum_j T_{ij} V_j - B_i) \tag{4.8}$$

where V_i is the output of a node, which is considered as the degree of confidence about an edge element presenting in pixel (x_i, y_i), and $f(\cdot)$ is a decision function, usually nonlinear. It is desired that, within the window, with a proper choice of parameters, isolated edge pixels should be removed and small gaps should be filled and the links T_{ij} of the ith neuron form a lateral inhibition function to give thin edges, when the orientation field in this window is homogeneous. Equation (4.8) however, only defines edge elements using information in a given window. The global energy function

$$E = -\frac{1}{2}\sum_i \sum_j T_{ij} V_i V_j - \sum_i B_i V_i \tag{4.9}$$

is used for the selection of edges. It is not hard to see that in an ideal case, edges of length at least 2 elements are preferred, for such a configuration will reduce the term $-T_{ij}V_iV_j$ in the energy function. The bias term B_iV_i can be used to introduce some *a priori* knowledge to regularize the edge configurations.

Thus far, edge detection has been formulated as an optimization problem. The problem on hand is how to determine the compatibility function and to search for

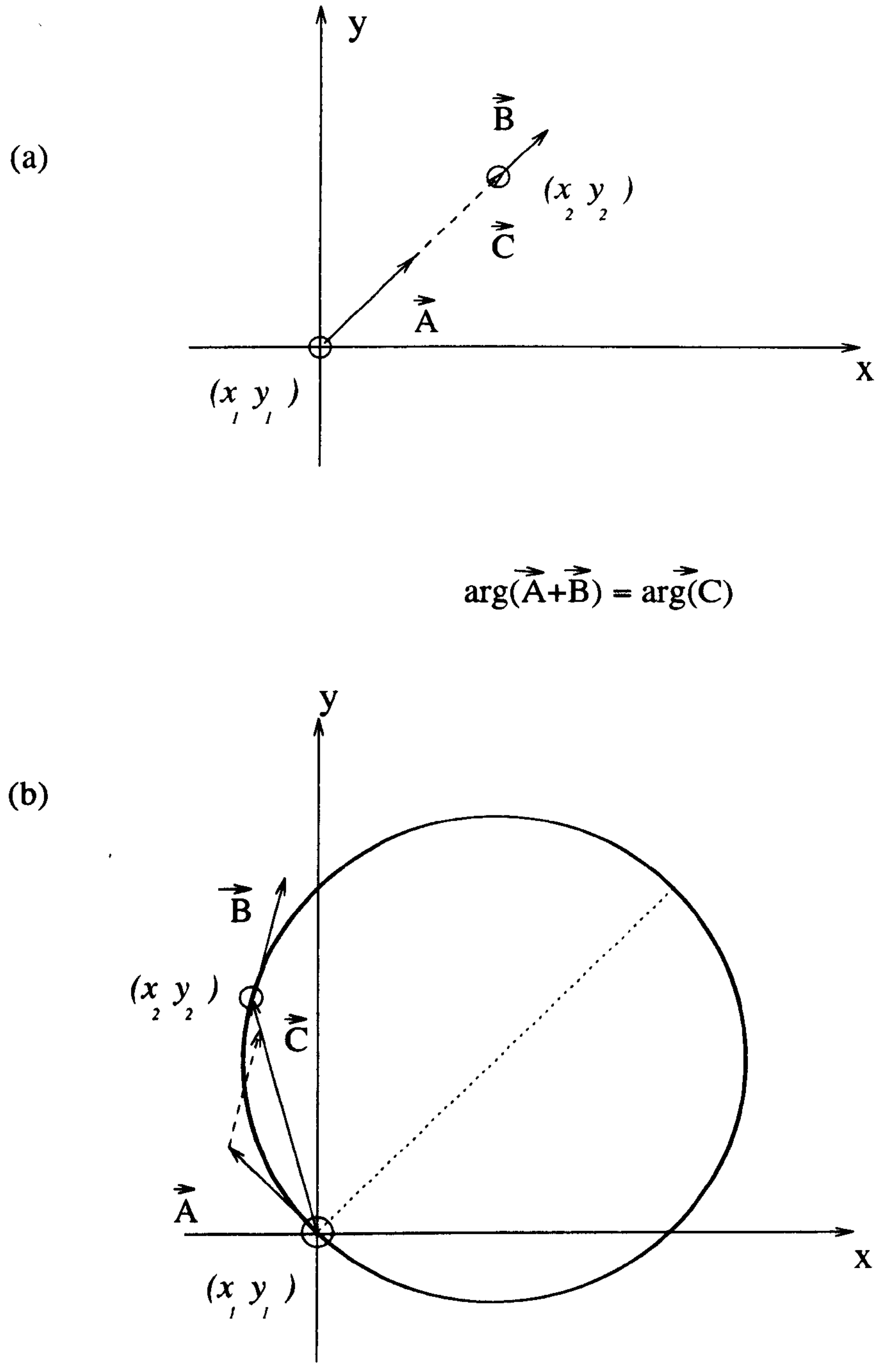


Figure 4.1: The setting of reinforcement links

the solution. The compatibility measure between two pixels should reflect the idea that when two pixels are in the same shape, the measure is high; otherwise it should be low. A good measure for two pixels' membership in a shape, or 'co-shape' is the angle difference between the displacement vector between them and the sum of the two orientation vectors of the two pixels. This is illustrated in Fig 4.1. In Fig 4.1 (a), a straight edge passes through pixels (x_1, y_1) and (x_2, y_2). The angle of the vector sum of the orientation vector $\vec{A}$ and the orientation vector $\vec{B}$ is the same to that of the displacement vector $\vec{C}$, which is from (x_1, y_1) to (x_2, y_2). Figure 4.1(b) shows that in an ideal circle, the magnitude of orientation vector $\vec{A}$ of pixel (x_1, y_1) is the same as that of orientation vector $\vec{B}$ of pixel (x_2, y_2). and it is obvious that again, $\vec{A}$ and $\vec{B}$ is the same as that of coordinate vector $\vec{C}$. In other words,

$$arg(\vec{A} + \vec{B}) = arg(\vec{C}) \tag{4.10}$$

where $arg(\cdot)$ gives the angle of a vector. The compatibility measure underlies the collinear concept. In an ideal case, this is similar to the concept of *co-circularity* of Zucker [143], but does not take curvature into account explicitly.

To ensure that the resultant edge is as thin as possible, the compatibility measure should give lateral inhibition, so that the winner of a competition between pixels across the direction of the edge is selected as the edge. A good candidate for both the measure of 'co-shape' and the competition is the second derivative of a Gaussian function. Since the interactions between pixels are local, distance should be also taken into account, that is, the measure should be proportional to distance between two pixels. A reasonable choice is a lowpass-like weighting function like the $cos(\cdot)$ function. Finally, the magnitude of the orientation vector also plays an important role: the more certain is the orientation, the stronger should be the link.

Hence, let $\vec{v_i}$ be the orientation estimate at pixel (x_i, y_i); the compatibility between two neurons i and j which correspond to pixel (x_i, y_i) and pixel (x_j, y_j) will be measured as follows

$$T_{ij} = k(\sigma^2 - b_{ij}^2)exp(-b_{ij}^2/2\sigma^2) \cdot cos(2\pi a_{ij}/(cw-1)) \cdot \mu_{ij} \tag{4.11}$$

where k and c are constants, w is the window size, μ_{ij} is the magnitude of $\vec{v_i} + \vec{v_j}$, σ depends on w (in this work, σ is 0.7) and

$$a_{ij} = \gamma_{ij} cos(\theta_{ij})$$

$$b_{ij} = \gamma_{ij} sin(\theta_{ij})$$

where

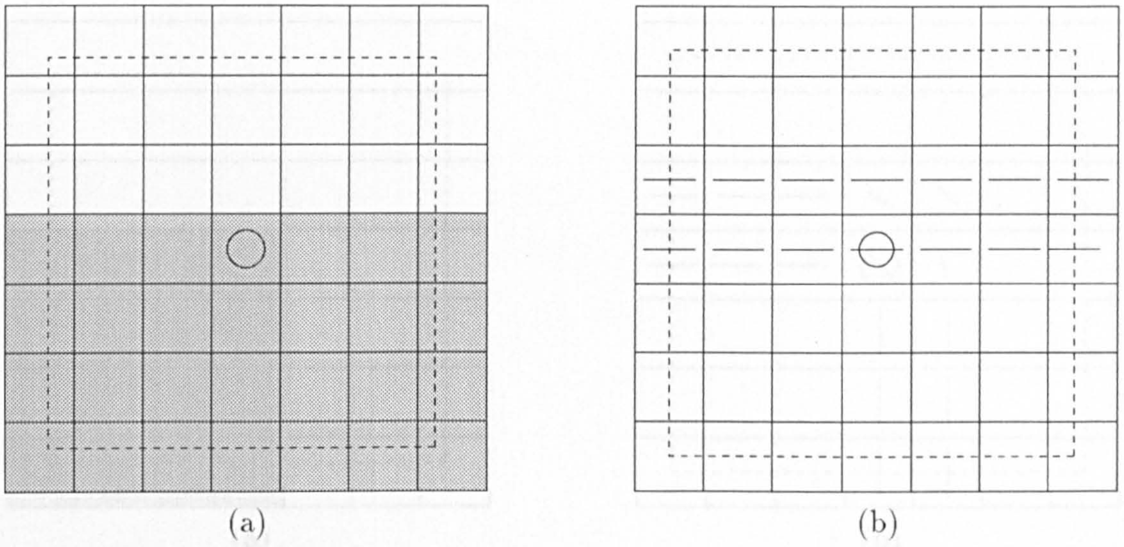
$$\theta_{ij} = \frac{1}{2}[arg(\vec{v_i} + \vec{v_j}) - 2 \cdot arg((x_i - x_j, y_i - y_j))]$$

$$\gamma_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

In other words, the compatibility between two edge components is separable into a function perpendicular to the edge, expressing lateral inhibition and one parallel to the edge, expressing continuity and proximity. To give an idea how the links setting will be, figure 4.2 shows the link strengths in a 5 × 5 window oriented horizontally and figure 4.3 and 4.4 are the link strengths of a right angle corner and 45 degrees acute angle corner. Note how this shows excitation along the feature orientation and inhibition in the perpendicular direction in figure 4.2. Unfortunately, the positive strengths do not follow corners exactly as shown in figure 4.3 and figure 4.4. This causes some defects in corners, which will also be discussed in section 4.5. Given various states of its neighbour nodes, figure 4.5(b) shows the energy of a node with the link settings shown in figures 4.2, 4.3 and 4.4. It is not hard to see that when the network output configuration corresponds to the orientation which underlies the link settings of the marked node, the energy of the marked node will be large. It is maximal for the line and right angle configurations, but for the acute angle, which fits the underlying collinear model least well, the largest energy is associated with a straight line, not the corner.

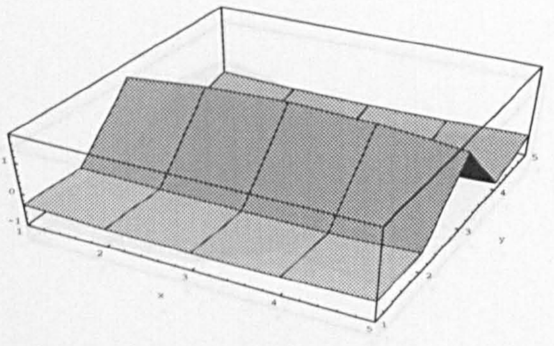
Having defined the compatibility function between two pixels, a technique to search the solution space is required. This is done by mapping the cost function to the energy function of a Hopfield neural net and the compatibility function to the link strength between two nodes. Some modification of the original Hopfield neural net, however, is required, for in this version there will be a self-feedback link for each node. The self-feedback, however, is set to be positive so that the net will converge. It is then necessary to check the change of the energy function each time a node is updated [97]. The implication is that the net will need extra computation for energy checking, which is, of course, a drawback. Nevertheless, with careful choice of parameters, it may be possible to avoid the checking [97].

The Hopfield network is applied to two clean images, 'table' and 'boats', to demonstrate its capacity(see figure 4.6 and figure 4.7). Figure 4.6(b) and figure 4.7)(b) are



(a)

(b)

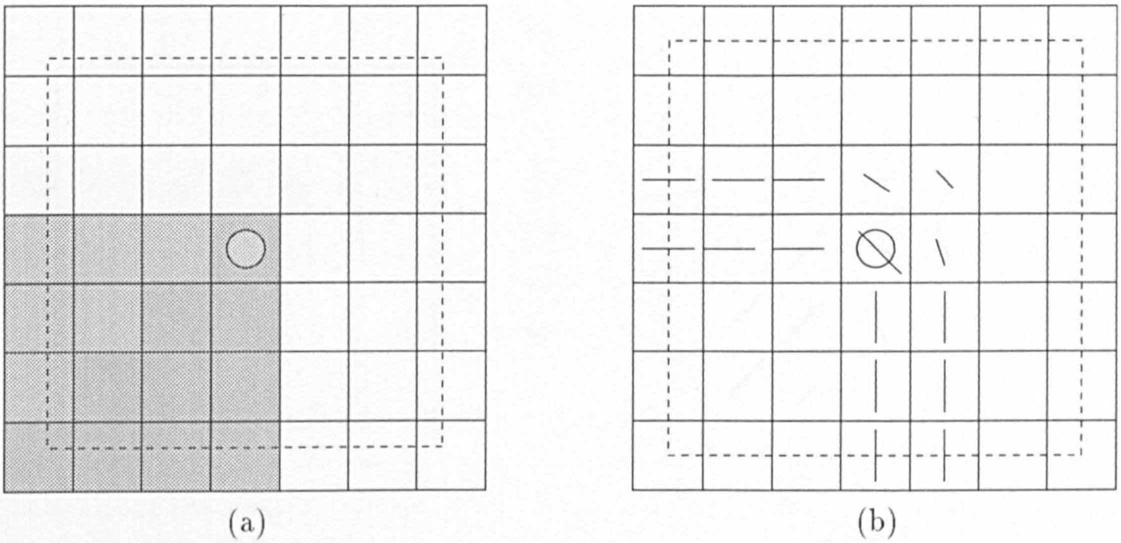


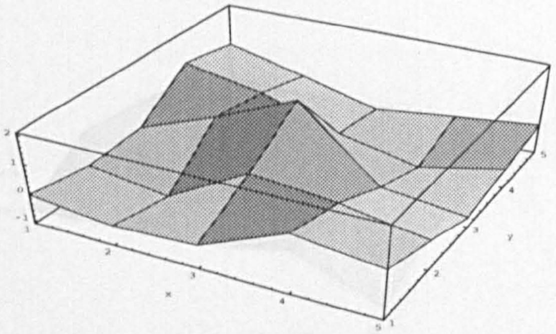
(c)

	x				
	-0.364	-0.428	-0.450	-0.428	-0.364
	-0.782	-0.919	-0.966	-0.919	-0.782
y	1.463	1.720	1.809	1.720	1.463
	-0.391	-0.459	-0.483	-0.459	-0.391
	-0.364	-0.428	-0.450	-0.428	-0.364

(d)

Figure 4.2: The link setting for a step edge.(a) An ideal step edge. (b) Its orientation. (c) A plot of the link strengths between a pixel and its neighbours. (d) The link strengths between this pixel and its neighbours.

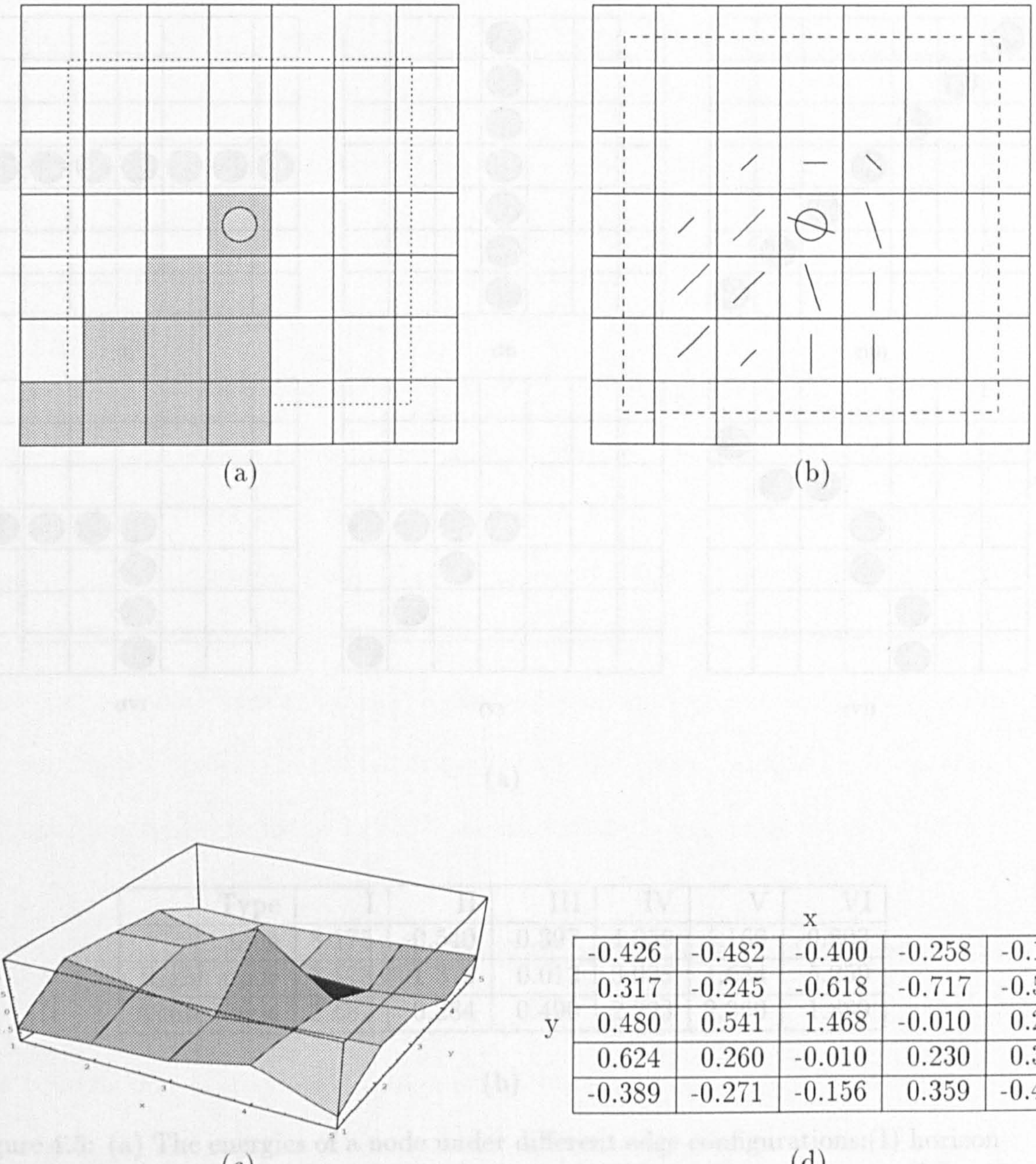




	x				
	0.630	-0.011	-0.733	-0.397	-0.073
	1.000	0.490	-0.413	-1.039	-0.397
y	-0.250	0.769	2.000	-0.413	-0.733
	-0.397	-0.812	0.769	0.490	-0.011
	-0.073	-0.397	-0.250	1.000	0.630

(d)

Figure 4.3: The link setting for a right angle corner. (a) A right corner of an ideal step edge. (b) Its orientation. (c) A plot of the link strengths between this corner and its neighbours. (d) The link strengths between this corner and its neighbours.



(a)

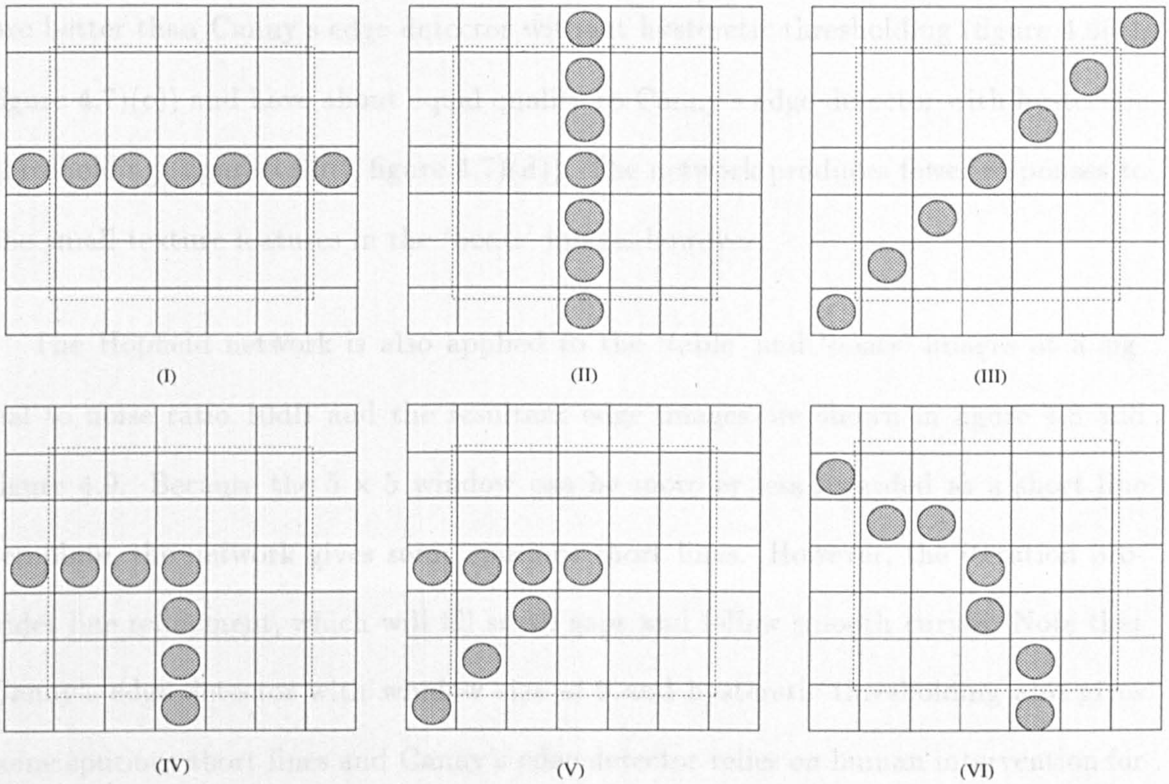
(b)

(c)

		x			
	-0.426	-0.482	-0.400	-0.258	-0.132
	0.317	-0.245	-0.618	-0.717	-0.510
y	0.480	0.541	1.468	-0.010	0.203
	0.624	0.260	-0.010	0.230	0.317
	-0.389	-0.271	-0.156	0.359	-0.426

(d)

Figure 4.4: The link setting for an acute angle corner(a) A 45 degree corner of an ideal step edge. (b) Its orientation. (c) A plot of the link strengths between this corner and its neighbours. (d) The link strengths between this corner and its neighbours.



(a)

Type	I	II	III	IV	V	VI
Line	8.175	-0.540	-0.397	4.059	4.169	-0.803
Right angle	1.473	1.373	0.013	3.038	1.634	5.259
Acute angle	2.682	0.284	0.490	2.223	2.360	1.889

(b)

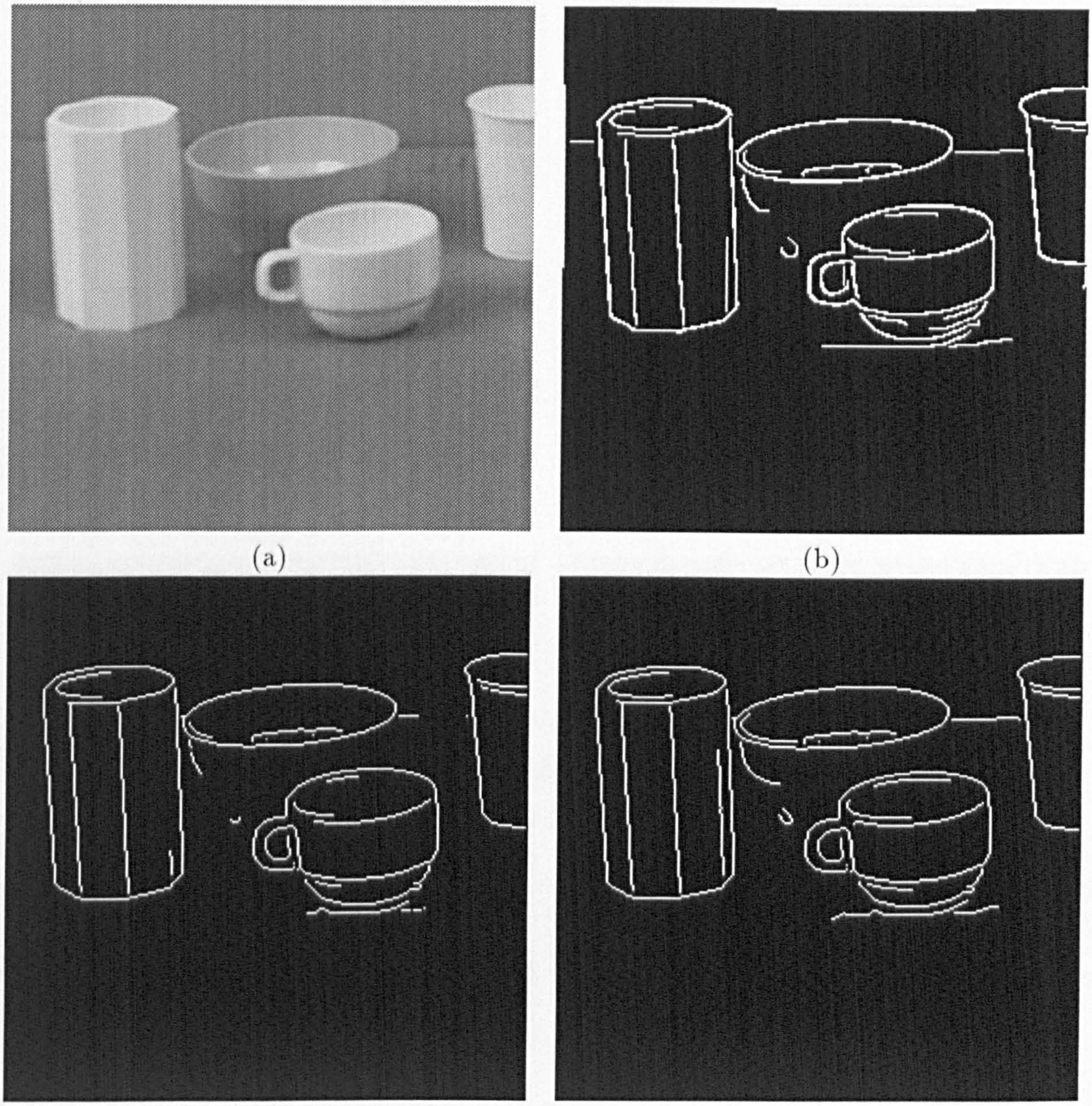
Figure 4.5: (a) The energies of a node under different edge configurations:(I) horizontal edge, (II) vertical edge, (III) diagonal edge and (IV) a right angle corner (V) a 45 degree corner (VI) a curve. Note the cells with shaded circles have magnitude 1 and cells without shaded circles have magnitude 0. (b) This table shows the responses of the node in three different link settings, which are shown in figure 4.2, 4.3 and 4.4.

the resultant edge maps obtained by applying the Hopfield network to the test images. Compared with Canny's edge detector followed by a thinning process [20], the results are better than Canny's edge detector without hysteretic thresholding (figure 4.6(c), figure 4.7)(c)) and have about equal quality to Canny's edge detector with hysteretic thresholding (figure 4.6(d), figure 4.7)(d)). The network produces fewer responses to the small texture features in the 'boats' image, however.

The Hopfield network is also applied to the 'table' and 'boats' images at a signal to noise ratio 10dB and the resultant edge images are shown in figure 4.8 and figure 4.9. Because the 5 × 5 window can be more or less regarded as a short line template, the network gives some spurious short lines. However, the iteration provides line refinement, which will fill small gaps and follow smooth curves. Note that Canny's edge detector with window size of 9 and hysteretic thresholding also gives some spurious short lines and Canny's edge detector relies on human intervention for choosing its parameters. In the following section, the above formulation is combined with multiresolution techniques to give an efficient and robust algorithm.

4.4 Orientation Estimation and Edge Detection

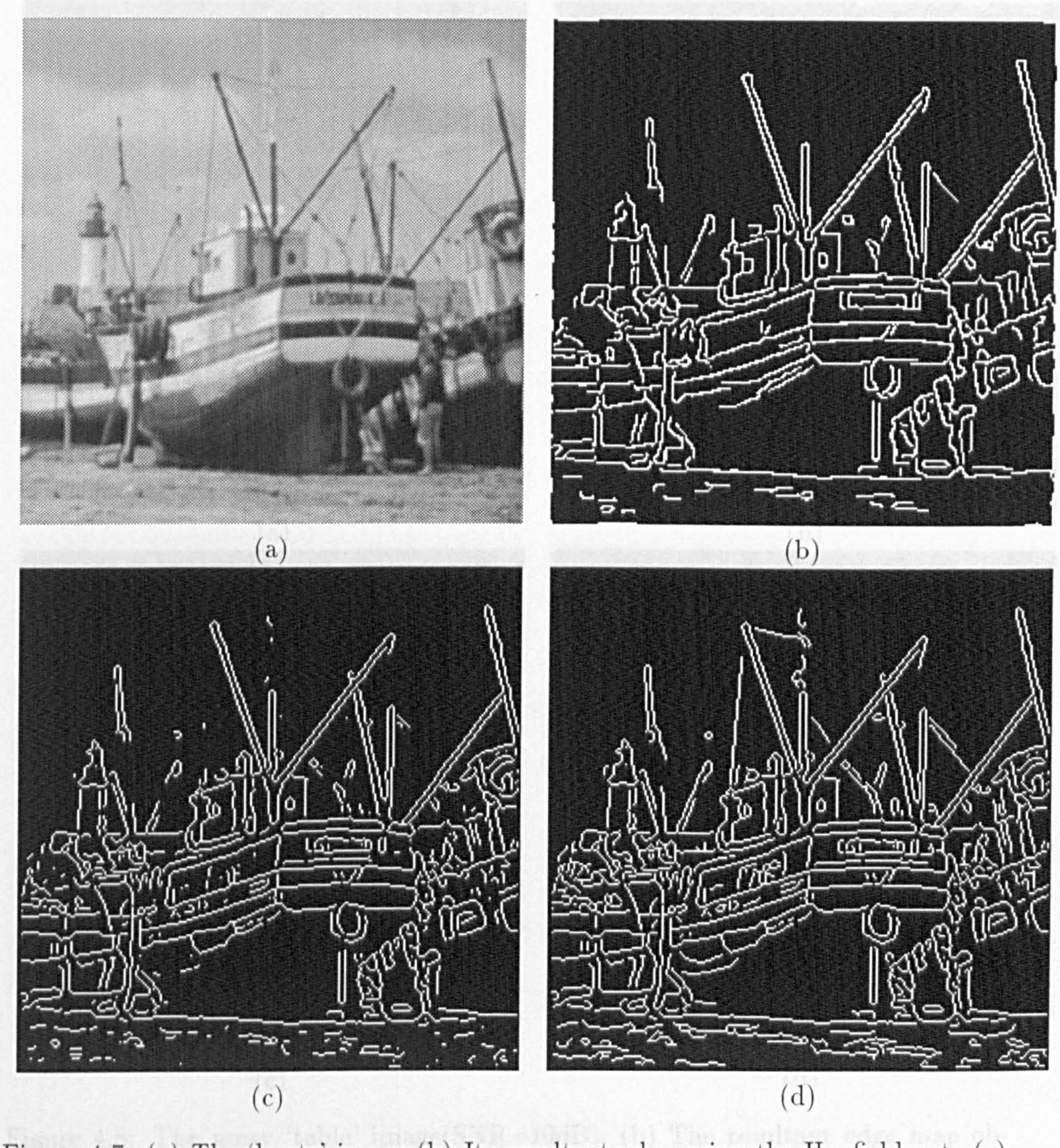
The construction of a suitable compatibility measurement enables the mapping of the edge grouping problem into an energy minimization problem. The relaxation technique has the same principle and has long been used for edge detection [143]. Using a stochastic relaxation scheme, Geman and Geman [35] proposed applying a line process to edges. Their line process idea was adopted by Zerubia and Chellappa



(a) (b)

(c) (d)

Figure 4.6: (a) The 'table' image (b) the resultant edge obtained using the Hopfield network. (c) Canny edge detector without hysteretic thresholding (threshold value = the magnitude below which 90% pixels lie), window size 9 pixels (d) Canny edge detector with hysteretic thresholding (high threshold value = the magnitude below which 90% pixels lie, low threshold value = 1/2 high threshold value), window size 9 pixels



(a)

(b)

(c)

(d)

Figure 4.7: (a) The 'boats' image (b) Its resultant edge using the Hopfield net. (c) The resultant edge map from Canny edge detector without hysteretic thresholding (threshold value = the magnitude below which 70% pixels lie), window size 9 pixels (d) The resultant edge map from Canny edge detector with hysteretic thresholding (high threshold value = the magnitude below which 70% pixels lie, low threshold value = 1/2 high threshold value), window size 9 pixels

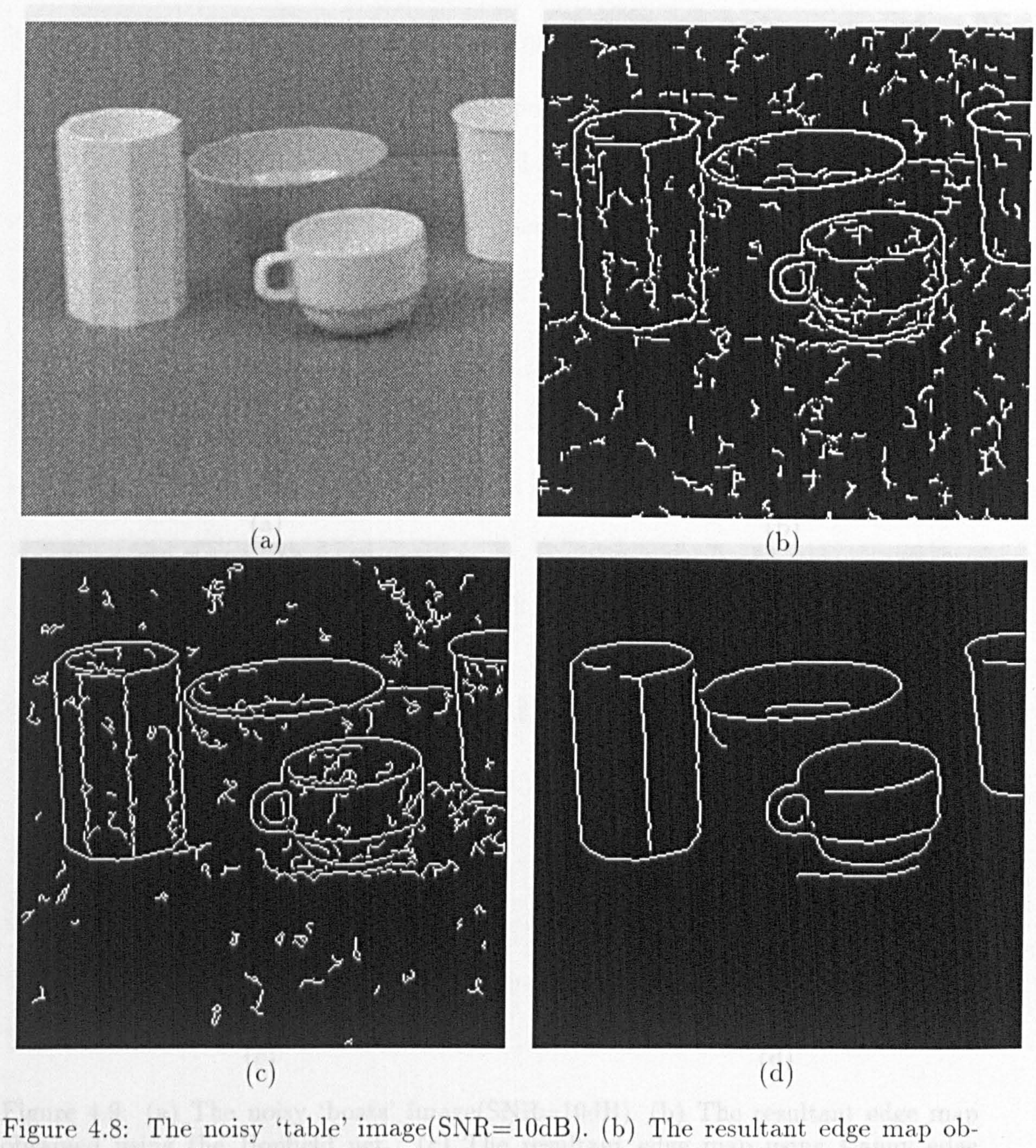


Figure 4.8: The noisy 'table' image(SNR=10dB). (b) The resultant edge map obtained using the Hopfield net. (c) The resultant edge map using Canny edge detector with hysteretic thresholding (high threshold value = the magnitude below which 70% pixels lie, low threshold value = 1/2 high threshold value), window size 9 pixels (d) The resultant edge map using Canny edge detector with hysteretic thresholding (high threshold value = the magnitude below which 70% pixels lie, low threshold value = 1/2 high threshold value), window size 17 pixels and deviation = 2 pixels.

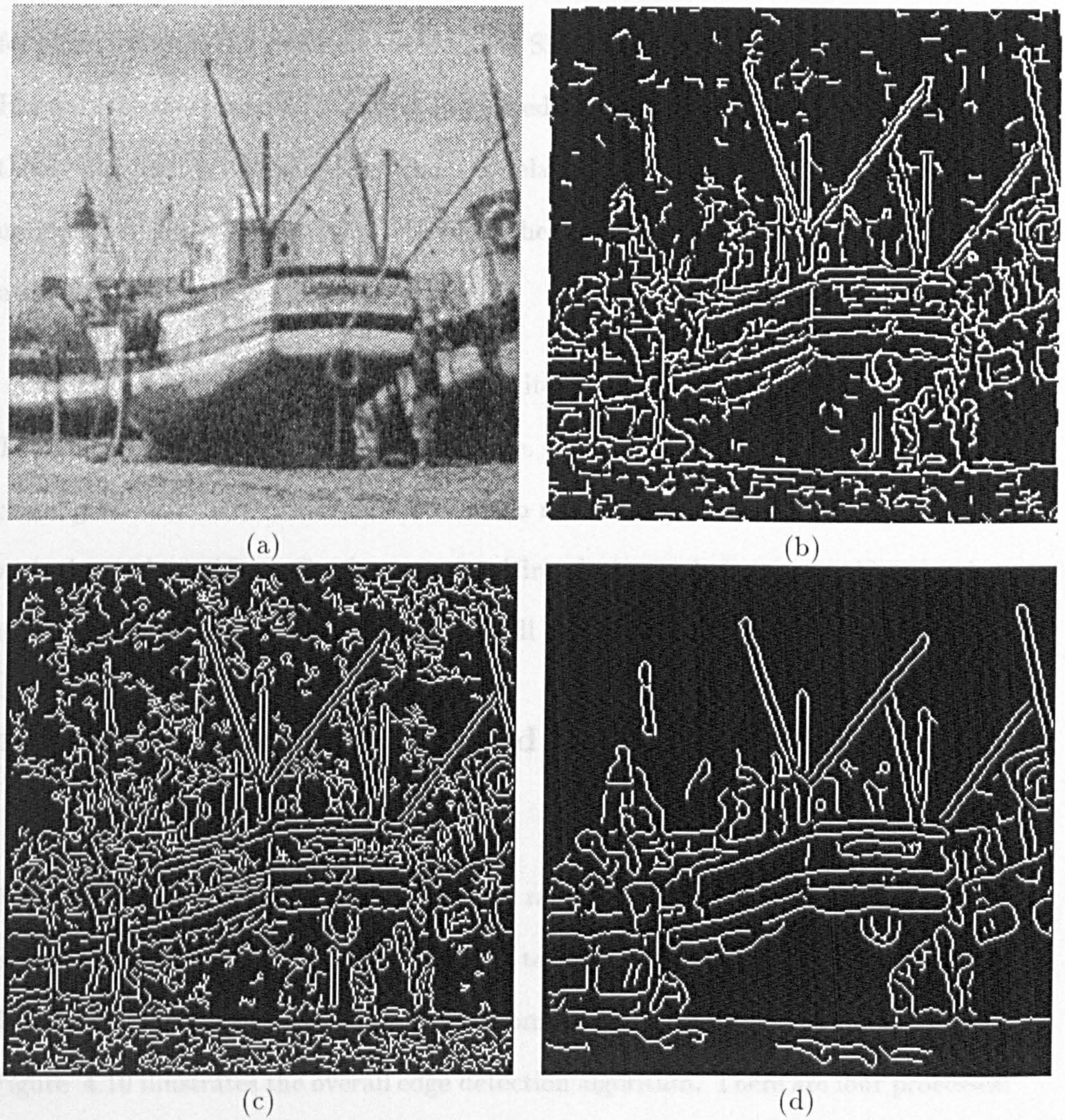


Figure 4.9: (a) The noisy 'boats' image(SNR=10dB). (b) The resultant edge map obtained using the Hopfield net. (c) The resultant edge map using Canny edge detector without hysteretic thresholding (high threshold value = the magnitude below which 70% pixels lie, low threshold value = 1/2 high threshold value), window size 9 pixels (d) The resultant edge map using Canny edge detector with hysteretic thresholding (high threshold value = the magnitude below which 70% pixels lie, low threshold value = 1/2 high threshold value), window size 17 pixels and deviation = 2 pixels.

[140] for edge detection. A related idea is to construct a locally connected network for edge detection, for example the works of Shaáshua and Ullman [116]. Herault and Horaud [45] also proposed a locally connected network, but they used the mean-field theory for finding the solution, which is related to the Ising model. The Hopfield network which is related to the mean field theory is easily implemented and is widely used, eg. [135].

All the techniques mentioned above use iterative schemes to search for solutions. The convergence rate is therefore an issue. A possible technique to improve the convergence rate is to use a multiresolution model[1], which also suitable for dealing with the scale problem. Furthermore, multiresolution techniques are able to reduce noise efficiently, so that the combination will give a robust edge detection scheme.

4.4.1 Multiresolution Hopfield Networks for Edge Detection

An image usually contains features over a range of scales. To deal with the scale problem, it is natural to use multiresolution techniques [88] [132]. The edge detection developed in the previous section can be combined with multiresolution techniques. Figure 4.10 illustrates the overall edge detection algorithm. There are four processes:

1. Construction of a lowpass gray level pyramid.

2. Initial orientation estimation.

3. Orientation directed edge detection using the Hopfield neural network.

[1] In solving differential equations, it is called the multigrid method

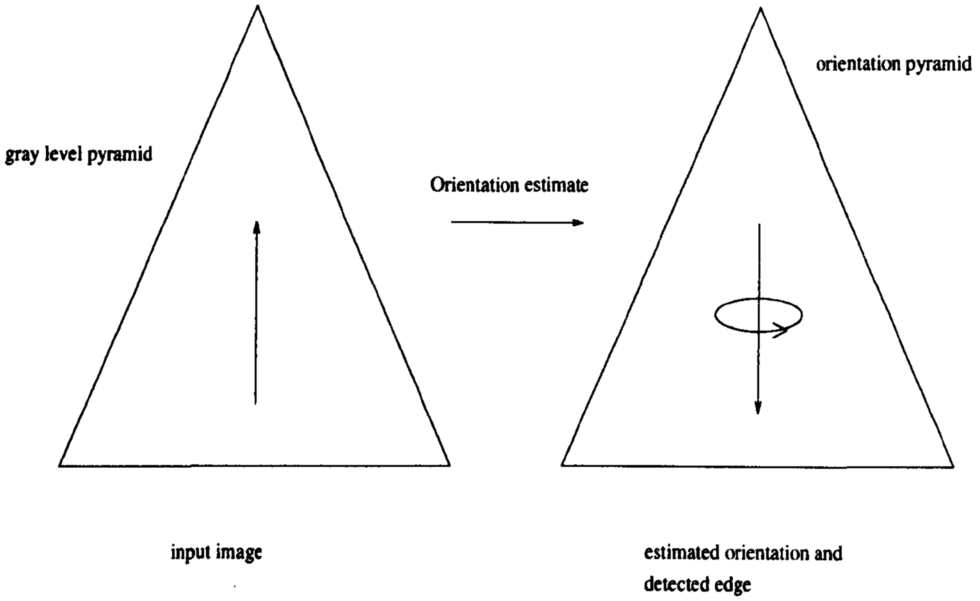


Figure 4.10: The structure of edge detection using a Hopfield network

4. Recursive top-down propagation of orientation estimates and detected edges.

Although construction of a gray level pyramid is a straightforward process, it is worth mentioning that a kernel with a circularly symmetric response is favoured, so that less orientation bias will be introduced to subsequent levels [128]. Given an image $f(i,j)$ and a kernel w_{mn}, the construction process is

$$f_{ij}(k) = \sum_m \sum_n w_{mn} f_{2i-m,2j-n}(k+1) \tag{4.12}$$

where $f_{ij}(k)$ is the image on level k , $f_{ij}(M) = f(i,j)$. Note that if the kernel size is even, the subsequent levels are shifted by one half pixel, as in a quad-tree.

To demonstrate the effect of the pyramidal smoothing, a synthesized image of shapes whose foreground grey level μ_1 and background gray level μ_2 are 170 and 150

-.0741	-.0955	0
-.0955	0	.0955
0	.0955	.0741

0	.0955	.0741
-.0955	0	.0955
-.0741	-.0955	0

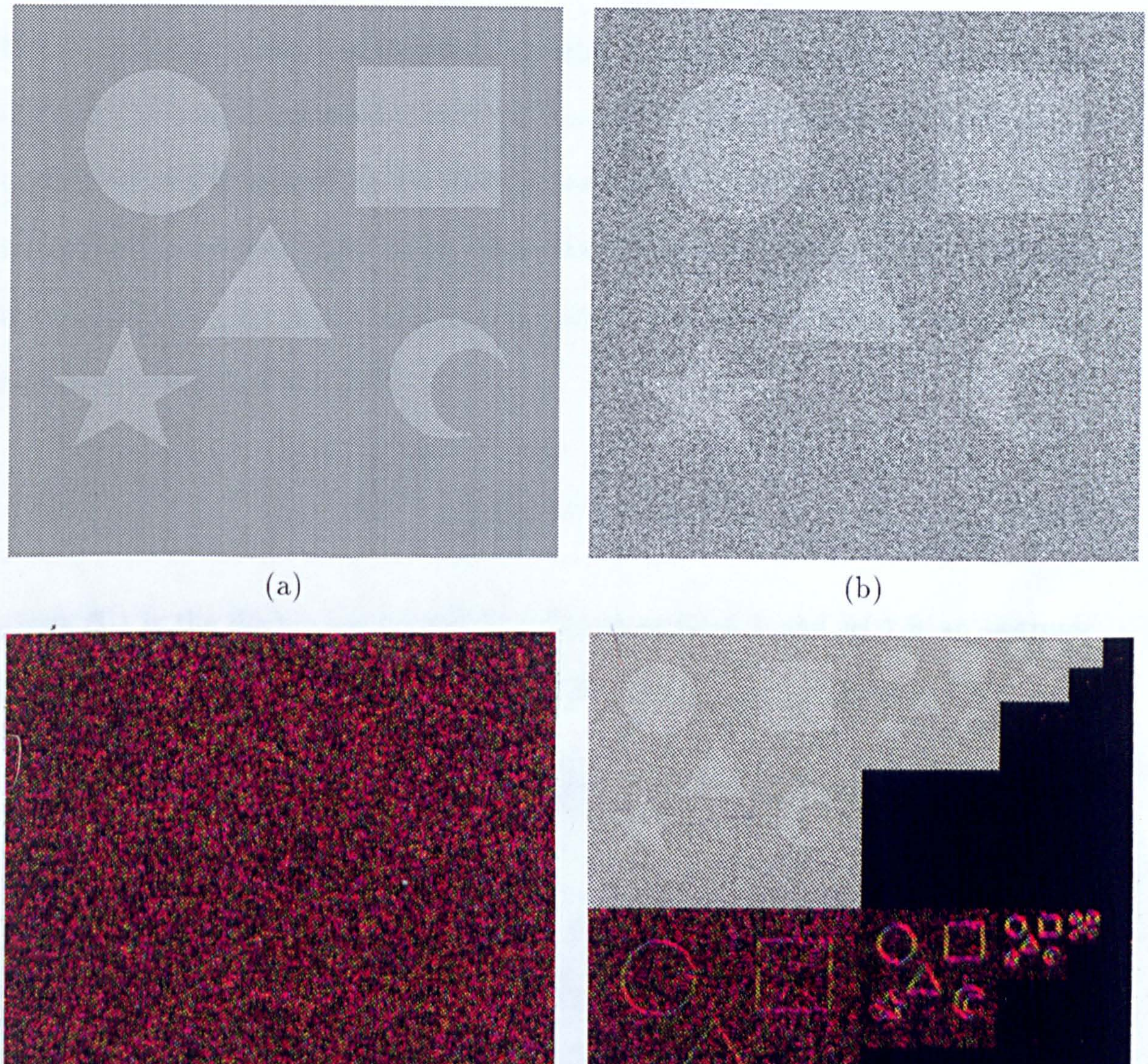
Table 4.1: Wilson's 3 × 3 edge detection kernels

respectively (see figure 4.11(a)) is used. Gaussian white noise with a variance of 400 is added to the synthesized 'shapes' image, giving an image whose signal to noise ratio is 0dB (figure 4.11(b)). The SNR for the 'shapes' image is calculated as

$$SNR = 10\log_{10}(\frac{\mu_1 - \mu_2}{\sigma_n})^2 \qquad (4.13)$$

Note that at the level of size 32 × 32, most noise is effectively removed by pyramid smoothing.

The construction of an orientation pyramid proceeds by convolving the image at each level with a pair of partial derivative kernels (Table 4.1). As described in section 4.3, the double-angle representation is adopted. These kernels are oriented at angles of $\frac{\pi}{4}$ and $-\frac{\pi}{4}$ respectively; thus a rotation of $\frac{\pi}{4}$ is applied so that the orientation field is properly aligned. Orientation estimation using the above 3 × 3 kernels is cheap and sufficient for images which consist of homogeneous regions separated by clear boundaries. However, when lines and texture edges are involved, more complicated filters should be used (see [67]). An orientation pyramid is shown in figure 4.11(c) and (d). Note that the double angle representation is used.



(a)

(b)

(c)

(d)

Figure 4.11: (a) The clean 'shapes' image (b) The 0dB 'shapes' image (c) Its orientation (d) The pyramid of the 0dB 'shapes' and its orientation pyramid. Note that hue corresponds to orientation and value (intensity) to magnitude in (c) and (d).

4.4.2 Coarse-to-fine Refinement

Having built the orientation pyramid, the edge detection scheme will start at a level in which the noise is removed or reduced to a negligible level by pyramid smoothing. A Hopfield neural network as described in section 4.3 is set for edge detection. After the network converges, those nodes which are 'on' are edge pixels. Hence, the image is classified into edge pixels and non-edge pixels. The orientation estimates at edge regions are enhanced as follows

$$\check{\vec{c}}_{ij}(k) = \sum_{n} \sum_{m} w_{i-m,j-m} \hat{\vec{c}}_{mn}(k) \tag{4.14}$$

where $\hat{\vec{c}}(\cdot)$ is the double angled vector estimate at level k and $w(\cdot)$ is an isotropic lowpass weighting function (cf. equation (4.7). For $w(\cdot)$, see [127]).

Now comes the problem of how to combine estimates of orientation information across scales. A characteristic in images that can be exploited for this purpose is that salient features tend to exist over more than one scale. Thus, a simple averaging between levels will emphasise consistent features and reduce uncorrelated noise. This is because the data in coarser levels is more reliable but has less detail, while data in finer levels is more detailed but less reliable.

To facilitate this coarse-to-fine propagation, the model originated by Clippingdale and Wilson [24] is adopted. The variation of orientation across scales is modelled in the form of a linear multiresolution equation

$$\vec{d}_{ij}(k+1) = \sum_m \sum_n A_{ijmn} \vec{d}_{\frac{i}{2}-m,\frac{j}{2}-n}(k) + \vec{\nu}_{ij}(k+1) \tag{4.15}$$

where $\vec{d}_{ij}(k+1)$ is the orientation vector of pixel (i,j) at level $k+1$, A_{ijmn} is a scalar interpolation function and $\vec{\nu}_{ij}(k+1)$ is an innovation vector. In its simplest form, the model is

$$\vec{d}_{i,j}(k+1) = \vec{d}_{\frac{i}{2},\frac{j}{2}}(k) + \vec{\nu}_{ij}(k+1) \tag{4.16}$$

which is used by Clippingdale [24] on a quadtree. Given the set of noisy data

$$\tilde{\vec{d}}_{ij}(k) = \vec{d}_{ij}(k) + \vec{n}_{ij}(k) \tag{4.17}$$

Clippingdale showed that the MMSE estimator $\hat{\vec{d}}_{ij}(k)$ for level k is a linear combination of the estimator a level above and the observed noisy data at level k as follows

$$\hat{\vec{d}}_{ij}(k+1) = \alpha(k)\hat{\vec{d}}_{\frac{i}{2}\frac{j}{2}}(k) + (1-\alpha(k))\tilde{\vec{d}}_{ij}(k+1) \tag{4.18}$$

The model, of course, compounds the refinement of the orientation estimate and the assumption of scale consistency. Its MMSE estimator, however, has a defect, due to being spatially invariant, which tends to blur important features such as lines and edges in images. A refined model also proposed by Clippingdale [23] is to use a spatially variant coefficient for the estimate

$$\hat{\vec{d}}_{ij}(k+1) = \alpha_{ij}(k)\hat{\vec{d}}_{\frac{i}{2},\frac{j}{2}}(k) + (1-\alpha_{ij}(k))\tilde{\vec{d}}_{ij}(k+1) \tag{4.19}$$

where $\alpha_{ij}(k)$ is a function of local and global signal to noise estimates. Bhalerao [10] further modified the estimator for pyramidal structures as

$$\hat{\vec{d}}_{ij}(k+1) = \alpha_{ij}(k) \sum_{m} \sum_{n} A_{ijmn} \hat{\vec{d}}_{\frac{i}{2}-m, \frac{j}{2}-n}(k) + (1 - \alpha_{ij}(k)) \tilde{\vec{d}}_{ij}(k+1) \qquad (4.20)$$

Using this model, the enhanced orientation estimate is propagated down, according to a linear equation of the form

$$\check{\vec{c}}_{ij}(k+1) = \sum_{mn} A_{mn} \check{\vec{c}}_{\frac{i}{2}-m, \frac{j}{2}-n}(k) \qquad (4.21)$$

where $(i, j, k) \in B(k)$ and

$$B(k) = \{(s, t, k+1) | (s-2p)^2 + (t-2q)^2 < 4, (p, q, k) = \text{edge pixel}\} \qquad (4.22)$$

The orientation estimates of those pixels in $B^c(k)$(complement of $B(k)$) are set to be zero. In other words, only those orientation estimates at edge pixels are propagated down. It is not hard to see that the new estimate of the orientation is a spatially variant linear combination of the propagated orientation field and the noisy observation, that is

$$\hat{\vec{c}}_{ij}(k+1) = \alpha_{ij}(k) \check{\vec{c}}_{ij}(k+1) + (1 - \alpha_{ij}(k)) \tilde{\vec{c}}_{ij}(k) \qquad (4.23)$$

where

$$\alpha_{ij}(k) = \begin{cases} 0 & (i,j) \notin B(K) \\ \frac{\sigma^2(k+1)}{\sigma^2(k+1)+\sigma^2(k)} & (i,j) \in B(K) \end{cases} \qquad (4.24)$$

where $\sigma^2(k)$ is the normalised noise variance. In other words, it is assumed that the signal on two levels is identical and the noise is uncorrelated. These conditions are close to being met in this case.

The estimate of $\sigma^2(k)$ is obtained with the assumption that those pixels which are labelled edges are signal plus noise and the rest is noise. The edge area from the level above is projected down to give an initial guess of the edge area(i.e $B(k)$) for the estimation of $\sigma^2(k)$. After the initial orientation field estimate is obtained, the Hopfield neural network is used to detect edges in level $k+1$. The resulting edge map at level k is used as a bias to regularize the solution and to reinforce scale consistency. The whole process proceeds recursively until the bottom of the pyramid is reached (see figure 4.10).

4.4.3 Network Dynamics

The original Hopfield network updates its nodes asynchronously [48]. Because of the self-feedback links, the Hopfield network must check the change of energy. A well-established property of the Hopfield network is that it will converge to a stable state when operating in a serial mode (update only one node at any time interval) with a symmetric nonnegative diagonal connection matrix [13]. When operating in full parallel mode(update all nodes at every time interval), the Hopfield network with a symmetric nonnegative diagonal connection matrix will converge to a cycle of length at most 2. This suggests a synchronous update, which is convenient for digital hardware implementation. Both updating methods have been tried in this work and synchronous update seems to give better results, although it is not clear why. The steps of the edge detection algorithm are summarised in the following pseudo code.

```
procedure       EdgeDetectionUsingHopfieldNet
    begin
start at level k of  the orientation pyramid
        while  (k<=M)
             begin
                 Using the orientation estimate to
                 initialise the Hopfield neural network.
                 While not converged
                 begin
                       run Hopfield neural network
                 end
                 re-estimate the orientation
                 propagate the orientation estimate
                 k = k+1
                 estimate signal to noise ratio at level k
                 combine the initial estimate and the propagated
                 estimate
                 set the father level as the bias input
             end
   end
```

4.5 Results and Discussion

The scheme described in section 4.4 has been implemented and the results are discussed here. To show the effect of this multiresolution scheme, it is applied to several images of different SNRs. Figure 4.12(a) shows the orientation field pyramid of the 'table' image which is shown in figure 4.6(a). The resultant edge map for each level is shown in figure 4.12(b). Compared with figure 4.6(b), several weak edges have been missed. This is because when an edge is weak and not classified as a feature, the combination of estimates between levels further weakens the edge information unless, at a certain level, the edge becomes salient and is classified as a feature. The problem is also illustrated in figure 4.14 and figure 4.15. Figure 4.14((c) and (d)) is the the result of applying the multiresolution Hopfield network to the 0 dB 'Barbara' image. The coarse-to-fine process starts from the level of size 64 while figure 4.14 is the result of starting from size 32. Obviously, there are certain edges missing in figure 4.14. Nevertheless, this problem is not severe, as it is noted that the model allows new features to emerge in the coarse-to-fine refinement of the edge pyramid and salient features tend to exist across several scales. Bearing in mind the simplicity of the processing at each scale and the robustness to noise, these results seem better then those reported elsewhere [45] [66]. Also shown in figure 4.13 and figure 4.16 are the edge maps of 'table' and 0dB 'Barbara' after iteration 1, 2, and 4 respectively. Apparently, the network recovers weaker edges segments in a process which is reminiscent of the associative memory.

For comparison, the multiresolution Hopfield network is applied to two noisy 'table' images. The resultant edge maps are shown in figure 4.17(b) and figure 4.18(b). Comparing figure 4.17(b) with figure 4.8, the noise is effectively reduced with the computation cost less than a factor of 2. Although there are also several edges missed, the result seems better than those from the Canny edge detector. The reader is reminded that the results of Canny edge detector rely on an appropriate choice of the thresholding values. The multiresolution scheme is robust even when the SNR of an input image is as low as 0dB. Note that in figure 4.18 the result from the multiresolution scheme is as good as those from a Canny edge detector with window size as large as 19.

Further to test the network's performance at corners when white noise is present, the synthesized image 'shapes' (see figure 4.11) is used. Figure 4.19 shows the edge detection result of SNR 10 dB 'shapes' image. Because the compatibility measure is designed for tracking straight lines and low curvature curves, its performance at corners is defective. When a corner is acute, the scheme will miss the corner because of the ambiguity of the orientation information at the corner area. As is shown, the corners of the crescent are not linked and the lower one is lengthened or overshot. This is because the corner is very sharp. Thus, the end point of the corner is excited by both lines, and due to lateral inhibition, those pixels of the inner edge (with a larger curvature) of a sharp corner will be turned off, as is seen in the result. If the corner is less acute, the network will try to follow a smooth curve which ends up with a round corner (see the corners of the square).

As noted earlier, the 5×5 kernel can be, more or less, viewed as a template of a short line. It thus resembles a template edge detection in the sense that it determines

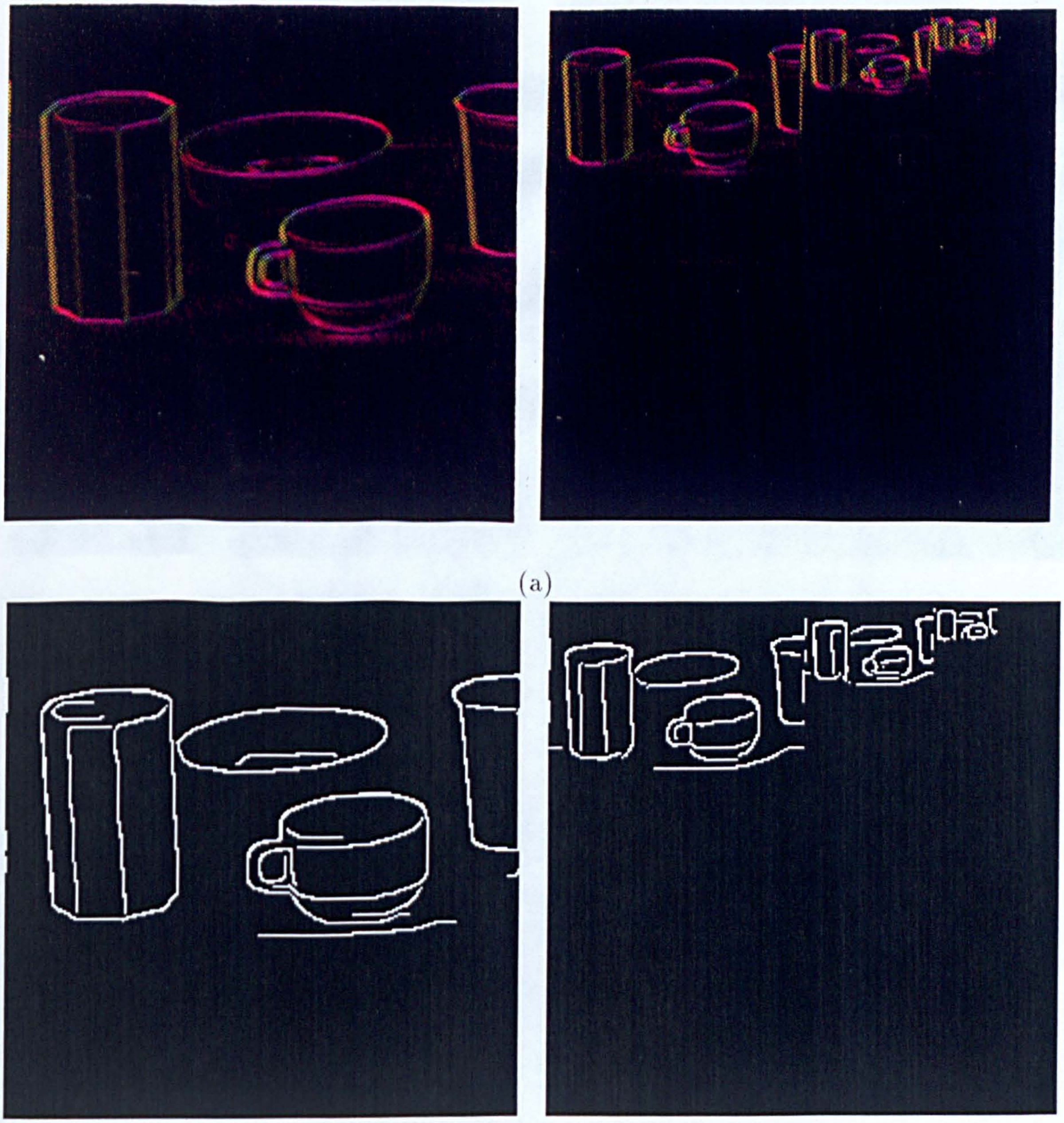
image	level 8	level 7	level 6	level 5
table	7	19	31	24
table (10dB)	7	8	22	13
table (0db)	6	18	13	18
Barbara(0dB)	10	15	15	18
shapes(0dB)	7	9	17	12

Table 4.2: The number of iterations in different levels(level 8 is the size 256)

a line in a small window. However, iteration of the network provides line refinement which a template edge detection lacks. Notice the edges in all the edge maps are one or two pixels wide. If an edge is too flat perpendicularly to its direction, the edge may have a ghost(see the figure 4.20). The problem is caused by the lateral inhibition links.

Finally, the multiresolution Hopfield network usually requires about 20 iterations to converge. Table 4.2 shows the number of iterations in different levels for the test images.

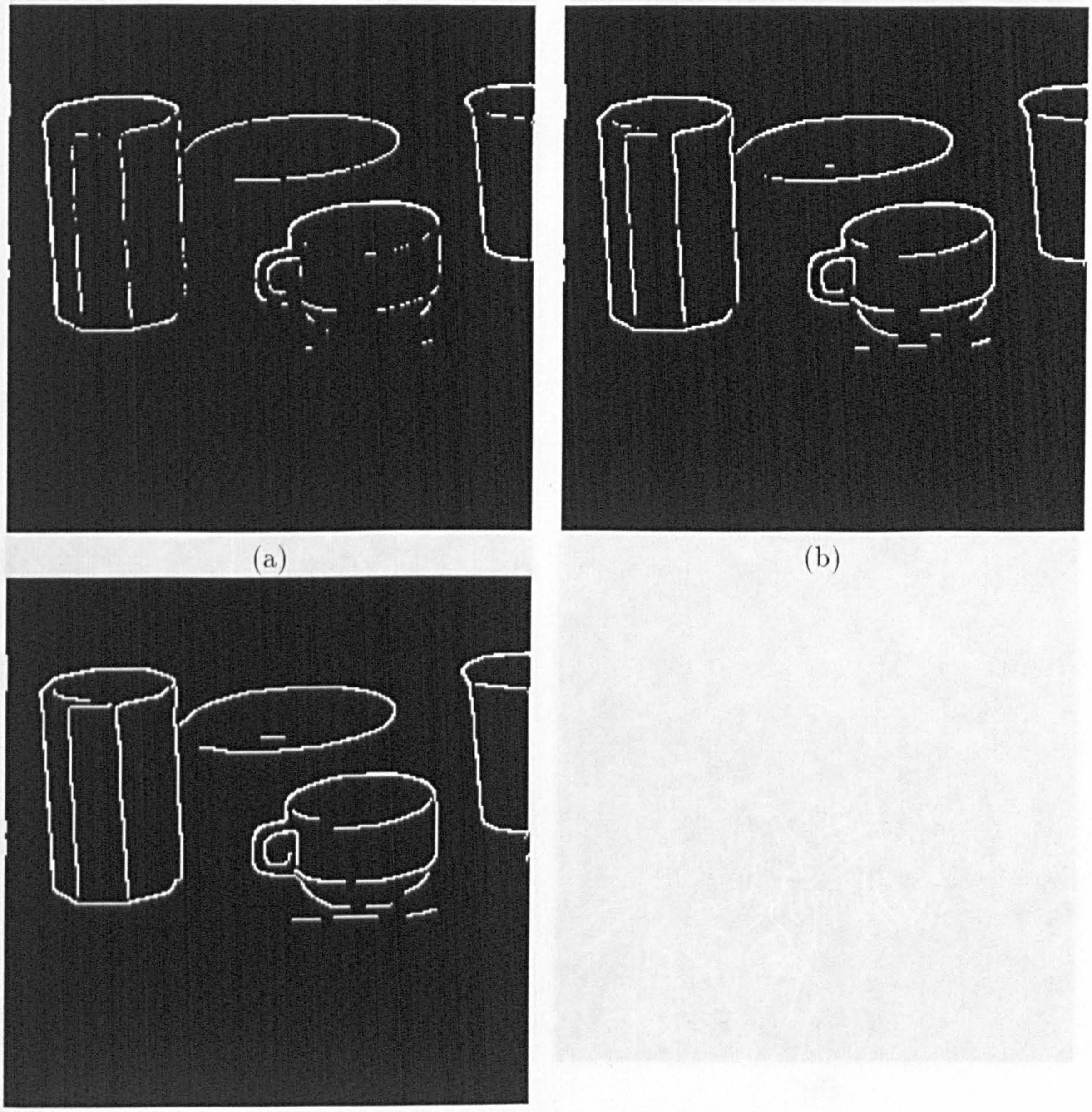
To sum up, the network provides (a) a simple local computation for edge grouping, (b) small gap filling is possible, (c) in general, only about 10-20 iterations are needed. The orientation estimation represents some improvement over techniques employing simpler enhancement strategies (eg. Clippingdale[23], Wilson *et al.*[128]). Nevertheless, it would appear that if more global context can be taken into account, (eg Bhalerao[10]) it may be possible further to improve the result. Moreover, the system is highly robust - no manual adjustment of parameters is required to get good results across a wide range of input signal-noise ratios. Finally, it shows that a network of simple neural units can indeed be used to produce an effective system for the extraction of boundary contours in a way which is amenable to parallel computation.



(a)

(b)

Figure 4.12: (a)(upper half)The orientation field of the 'table' image. (b)(lower half) the resultant edge map.



(a)

(b)

(c)

Figure 4.13: The edge maps of 'table' after (a) 1 iteration, (b) 2 iterations and (c) 4 iterations. The final result is shown in figure 4.12.

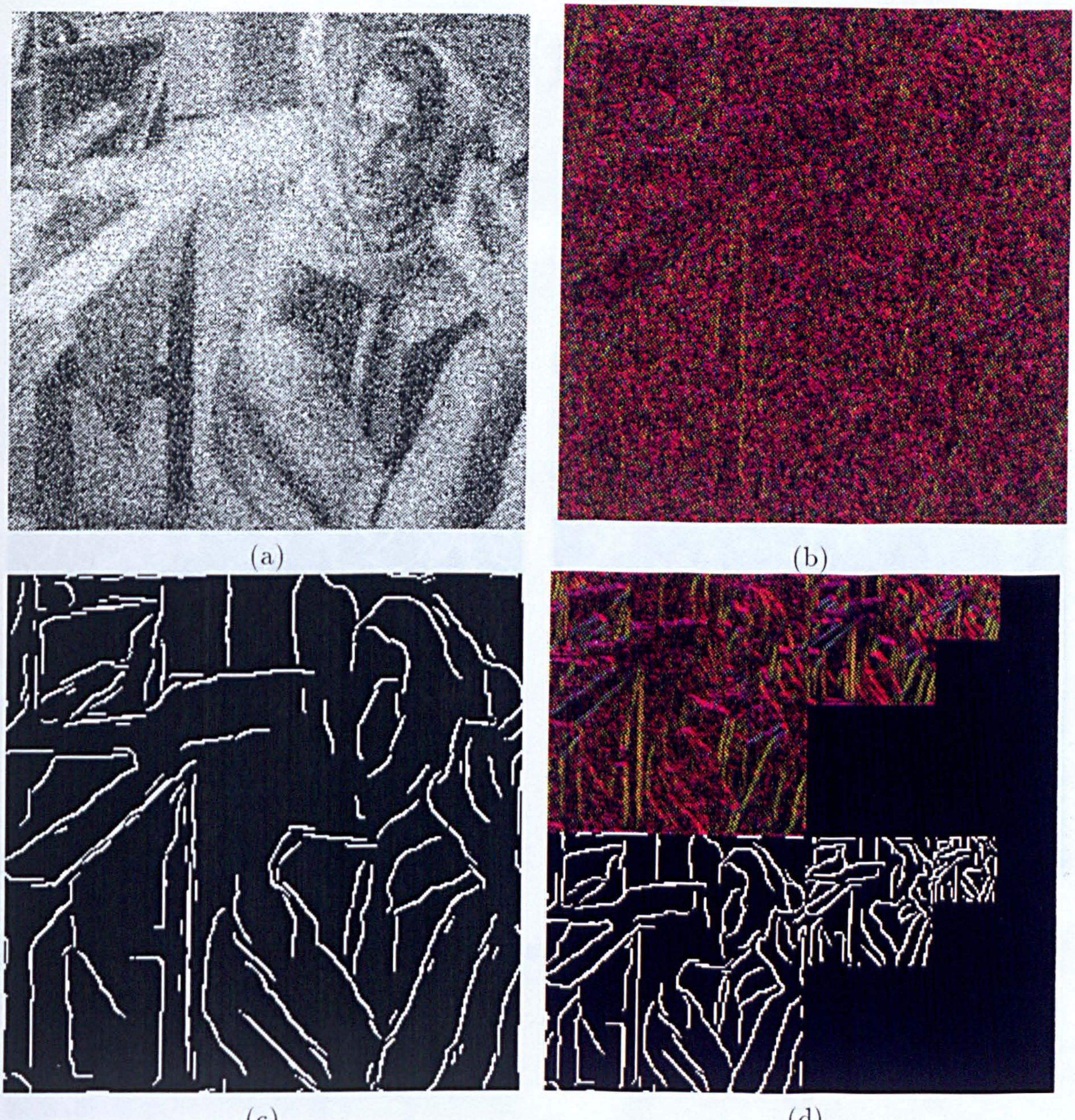
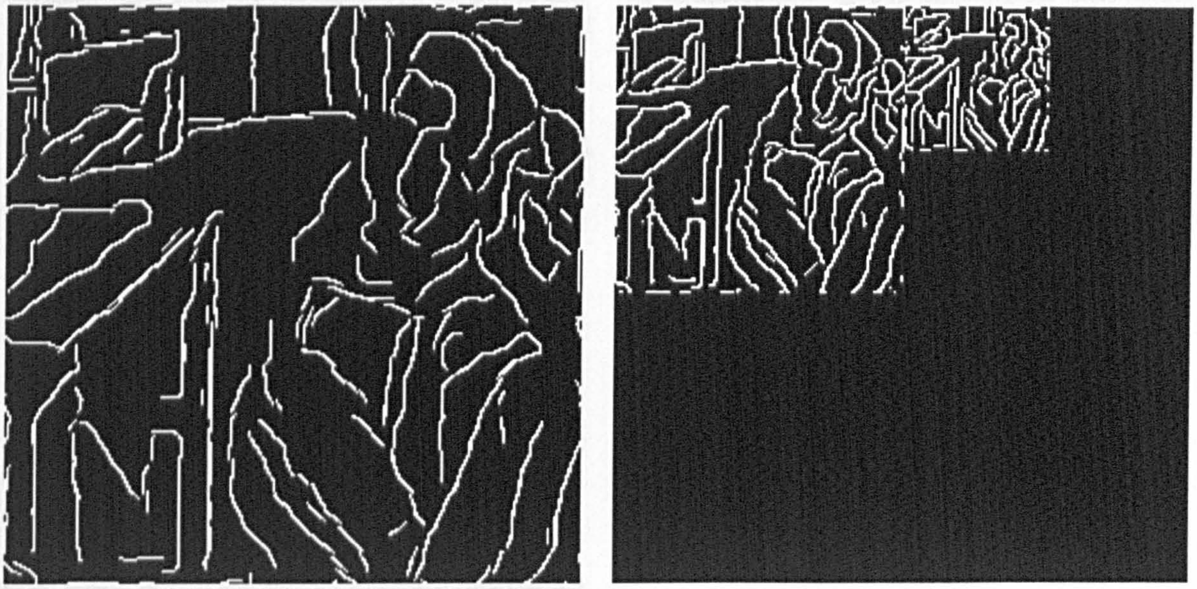
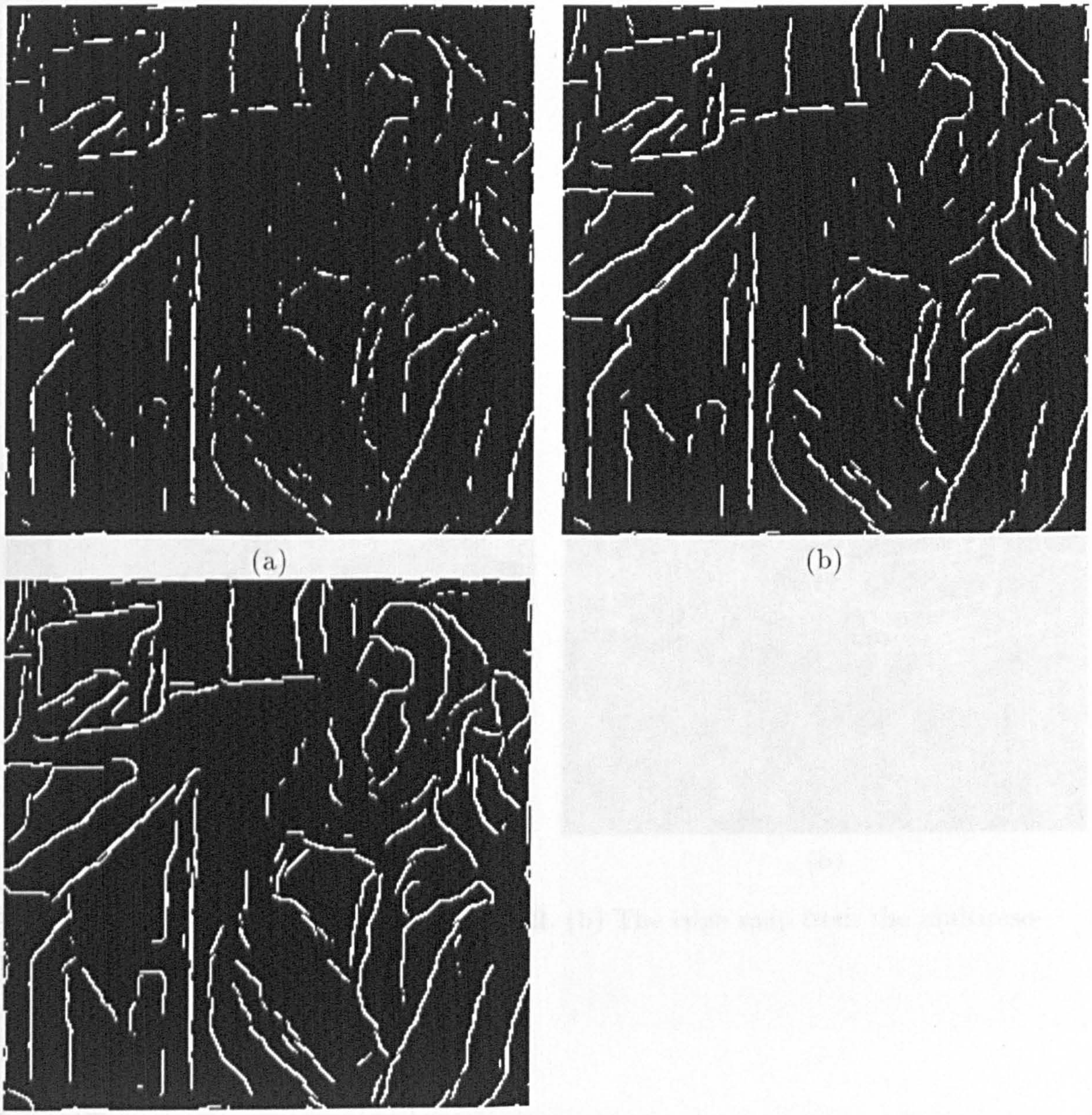


Figure 4.14: (a)The 'Barbara' image (SNR=0dB). (b)Its orientation field. (c)The resultant edge map. (d) Its double angled orientation field and edge map at different levels.



(a) (b)

Figure 4.15: The edge map of 0dB 'Barbara' (starting from the level of size 64).

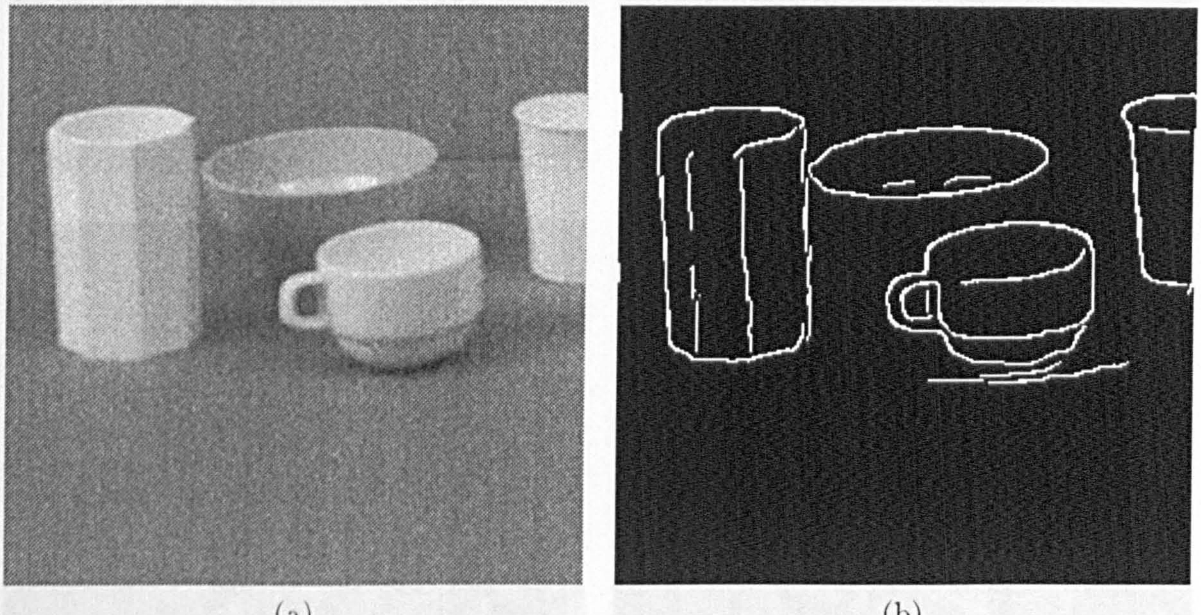


(a)

(b)

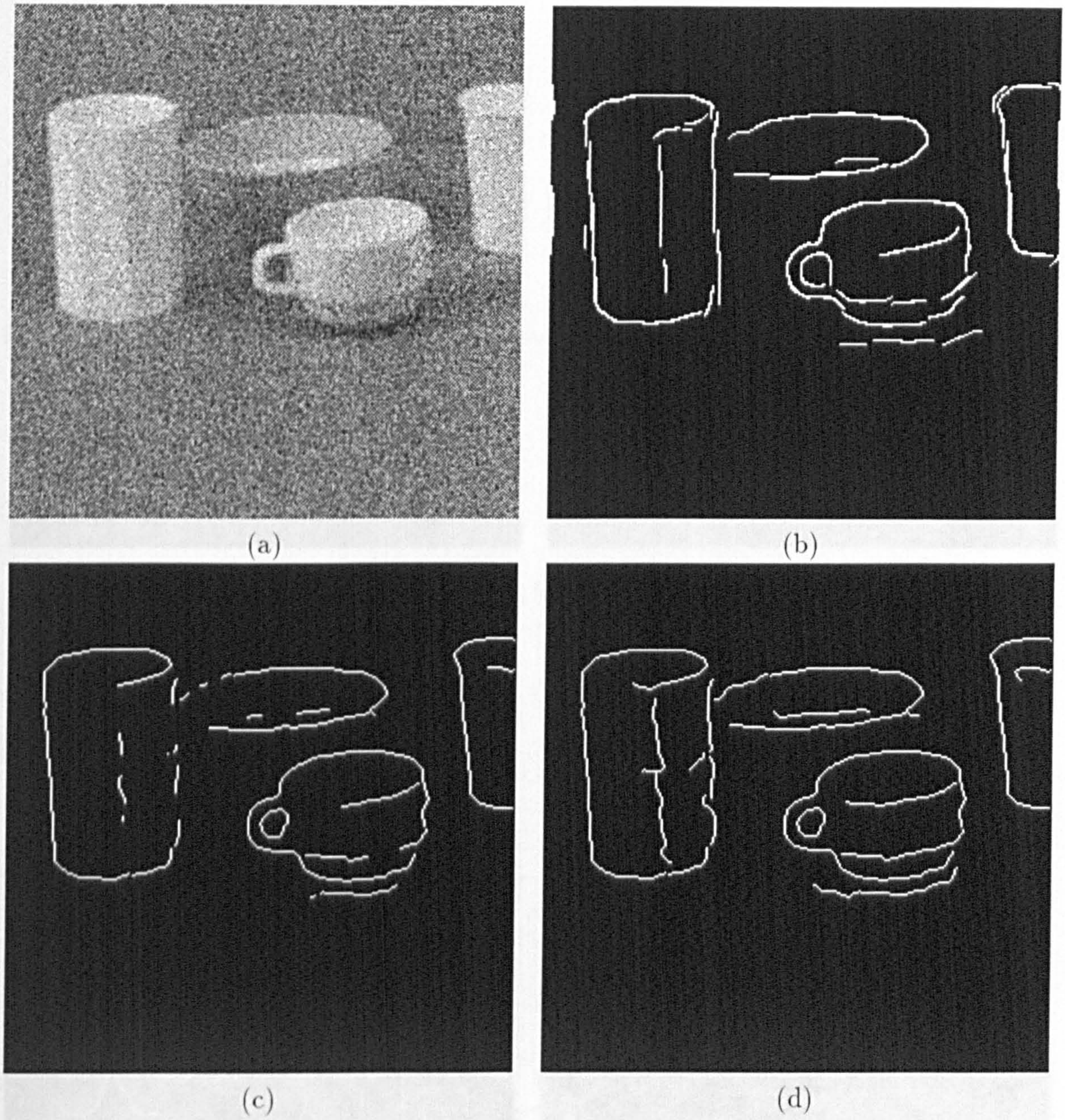
(c)

Figure 4.16: The edge maps of 0dB 'Barbara' after (a) 1 iteration, (b)2 iterations and (c) 4 iterations. (starting from the level of size 64) The final result is shown in figure 4.15.



(a) (b)

Figure 4.17: The 'table' image of 10dB SNR. (b) The edge map from the multiresolution Hopfield net.



(a)

(b)

(c)

(d)

Figure 4.18: (a) The 'table' image of 0dB SNR (b) The edge map from the multiresolution Hopfield net. (c) The edge map from Canny edge detector using sigma = 2.4 window size 19 and without hysteretic thresholding (threshold value = magnitude below which those of 90% pixels lie) (d)The edge map from Canny edge detector using sigma = 2.4 window size 19 and with hysteretic thresholding (high threshold value = magnitude below which those of 90% pixels lie, and low threshold value = 1/2 high threshold value).

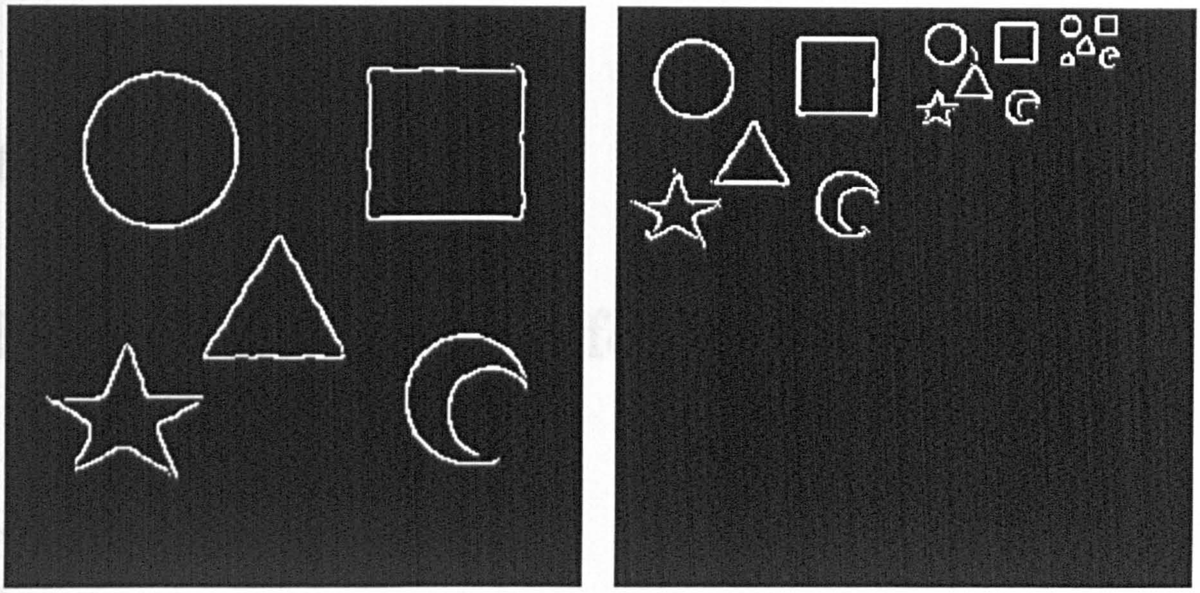


Figure 4.19: The edge map of the 'shapes' image (SNR=10dB).

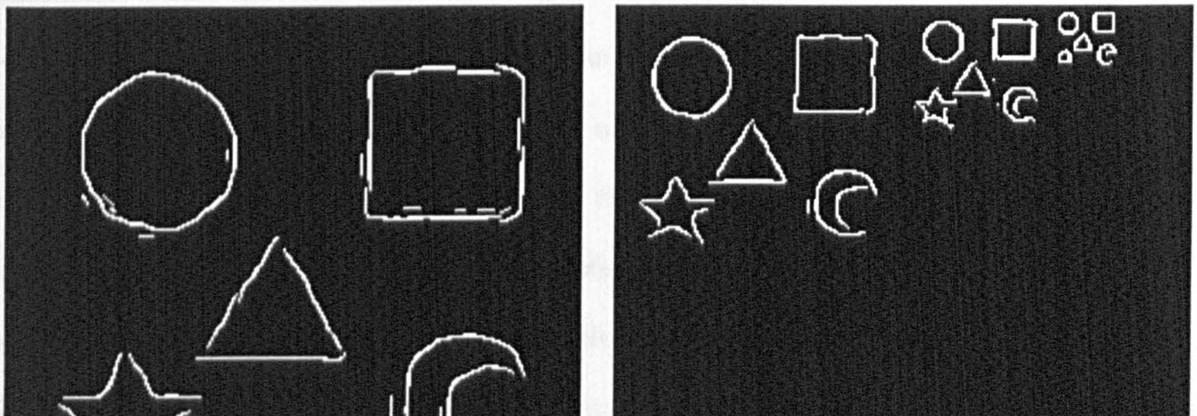


Figure 4.20: The edge map of the 'shapes' image (SNR=0dB).

Chapter 5

Neural Networks for Restoration

5.1 Introduction

The constrained least squares (CLS) method is a technique for restoring images degraded by blurring and additive white noise [109]. The CLS technique, which usually uses 'smoothness' as a constraint or a 'regularization term', is a form of regularization technique [104]. This method finds the solution which minimizes a quadratic error function. One way to find the solution is to use a quadratic error function as the energy function of a Hopfield network, so that the solution is obtained as the output of the Hopfield network [104][141]. In this chapter, a multiresolution Hopfield network is proposed for image restoration. The attraction of this approach is that the 'coarser' resolution solutions can be used in a regularization term and they can be also used to segment the images in the higher resolutions into regions of different activity, so that coarse-to-fine adaptive noise filtering is possible [23]. To further improve the performance of the multiresolution restoration scheme, an adaptive restoration scheme which uses the edge map obtained by applying the Hopfield network detailed in the previous chapter is also proposed.

5.2 Image Restoration and Regularization

An imaging system inevitably introduces some distortion into images due to turbulence of the environment and the physical limitations of the imaging system. Although these degradation processes are generally nonlinear, the distortions are often adequately modelled by a linear finite blurring plus additive Gaussain white noise, as follows,

$$y_{ij} = \sum_{(mn)\in\mathcal{N}} h_{mn} x_{i-m,j-n} + n_{ij} \tag{5.1}$$

where $\{h_{mn}\}$ denotes the point spread function of the blur, $\mathcal{N}$ is the support of the point spread function and n_{ij} is the additive white noise. Equation (5.1) can be written in a matrix notation,

$$Y = \mathbf{H}X + N \tag{5.2}$$

where $\mathbf{H}$ is the matrix representation of the blurring and Y, X and N are, respectively, the lexicographically ordered vector representations of the observed image, the original image and white noise. In this work, only linear shift invariant (LSI) blur is considered.

A classical method to restore the degraded image is to apply a Wiener filter to it. The Wiener filter is based on a stationary stochastic assumption, so that it does not adapt to salient features like lines and edges[78]. Alternatively, filters based on a statistical image model such as the Kalman filter, can be exploited [61]. This method, however introduces some bias in certain directions due to the causality requirement mentioned in chapter 2. Both Wiener and Kalman filters have been extended into adaptive versions [61].

If the degradation is only blurring, a straightforward method is to apply a matrix inverse operation to the observed image Y, which gives the source image $\hat{X} = \mathbf{H}^{-1}Y$ (or $\hat{X} = (\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}Y$ if $\mathbf{H}$ is singular or ill-conditioned). Although simple in theory, in practice the size of the matrix causes computational problems. Moreover, with the presence of noise, an inverse operation will also indiscriminately amplify white noise so that the restored image is overrun by noise. Instead of direct matrix inversion, iterative schemes are employed for searching the solution space, which minimize

$$E(X) = \| Y - \mathbf{H}X \|^2 \tag{5.3}$$

where $\| \cdot \|$ is a vector norm. It has been reported that for such methods, white noise is magnified proportionately to the number of iterations [11]. However, iterative algorithms under a cost minimization framework are suitable for including *a priori* knowledge as penalty functions to prevent the magnification of noise [65] [73]. In such approaches, an 'energy' function is designed so that it combines posterior constraints (image data Y) and prior constraints, or regularization terms, to stabilise the solution. It is also possible to include cost functions for discontinuities. Hence, iterative schemes can be readily extended for adaptive restoration.

As discussed in section 3.5.1, the problem of restoring a distorted image is ill-posed, in the sense that its solution is not unique or does not depend continuously on the data [104]. To solve this problem, *a priori* knowledge is used to restrict the solution space and reduce the instability of the solution so that the problem becomes a well-posed one. The standard regularization method for the image restoration problem is

$$E(X) = \| Y - \mathbf{H}X \|^2 + \lambda \| \mathbf{D}X \|^2 \tag{5.4}$$

where λ is the regularization parameter and $\mathbf{D}$ is, typically, a high pass filter, such as the Laplacian operator. The solution for this problem is the estimate $\hat{X}$ which minimizes the error function E, and can be found by solving the linear system

$$(\mathbf{H}^t\mathbf{H} + \lambda\mathbf{D}^t\mathbf{D})\hat{X} = \mathbf{H}^tY \tag{5.5}$$

The above is a deterministic solution. Alternatively the image X can be considered as a Markov field

$$X = \mathbf{L}V \tag{5.6}$$

where V is a zero mean white noise field. The optimal solution to the degradation model (equation 5.2) is the Wiener filtered data

$$\hat{X} = (\mathbf{H}^t\mathbf{H} + \frac{\delta_n^2}{\delta_v^2}\mathbf{L}^t\mathbf{L})^{-1}\mathbf{H}^tY \tag{5.7}$$

where δ_n^2 and δ_v^2 are the variance of white noise n_{ij} and signal v_{ij}. Putting

$$\lambda\mathbf{D}^t\mathbf{D} = \frac{\delta_n^2}{\delta_v^2}\mathbf{L}^t\mathbf{L} \tag{5.8}$$

it is obvious that the classical Wiener filter is a special case of the regularization equation (5.4) [36] [108]. Equation (5.4) expresses the trade-off between data fidelity, which is dictated by the term $||Y - \mathbf{H}X||^2$ and smoothness by $\| \mathbf{D}X \|^2$. The regularization parameter λ, which usually depends on the variance of noise in the image, is used to select the trade-off. A common technique for choosing λ is maximum likelihood, but recently cross validation has also been used [33].

The above regularization method is a quadratic method cost minimization problem, which can be solved using the Hopfield neural network [104][141]. The Hopfield neural network model has a connection to the Ising model, which was refined by Geman and Geman for restoration [35]. Based on the same coupled MRF model of Geman and Geman, Bedini and Tonazzini [8] used a Hopfield network to find the minimum of a parameter function, which corresponds to the MAP solution of the MRF model for image restoration. A Hopfield network was investigated by Zhou *et al.* [141], who proposed a modified Hopfield network for image restoration using equation (5.4) as the energy function. Paik and Katsaggelos [97] further investigated the convergence properties of the modified Hopfield network and explicitly incorporated set theoretic constants. All the above methods rely on a quadratic energy function. Using the same error function (5.4), a multiresolution Hopfield network is proposed in the next section.

5.3 Multiresolution Networks for Restoration

5.3.1 Hopfield Networks for Restoration

In this section, the work of Zhou *et al.* [141] is reviewed. Based on their work, a multiresolution Hopfield network is proposed for coarse-to-fine restoration, in which an adaptive noise filtering is easily implemented(see section 5.4).

To use a Hopfield network for image restoration, the first task is to represent a grey level image using nodes. Because of the binary nature of nodes in the Hopfield network, a total of $N^2 \times M$ mutually connected nodes is needed to represent a digital $N \times N$ image of maximum grey level M. These nodes are divided into $N \times N$ groups of size M. The grey level of each pixel is represented by the sum of the activities of its corresponding group. Let

$$V = \{v_{ik} : v_{ik} \in \{0, 1\}, 1 \leq i \leq N^2, 1 \leq k \leq M\} \tag{5.9}$$

be the set of neurons used for representing a digital image. Then, the pixel value x_{st} is represented by the sum of the activities of a group of nodes as follows

$$x_{st} = \sum_{k=1}^{M} v_{ik} \tag{5.10}$$

where $i = (s - 1) \times N + t$.

Let T_{ikjl} be the connection weights between the node (i, k) and node (j, l). The energy function of the Hopfield net is then:

$$E = -\frac{1}{2} \sum_{i,j=1}^{N^2} \sum_{k,l=1}^{M} T_{ikjl} v_{ik} v_{jl} - \sum_{i=1}^{N^2} \sum_{k=1}^{M} B_{ik} v_{ik} \tag{5.11}$$

where B_{ik} is the bias input.

The activity of a neuron is determined by its input and its bias. Each neuron receives an input (cf. equation (4.1), (4.2))

$$u_{ik} = \sum_{j=1}^{N_2} \sum_{l=1}^{M} T_{ikjl} v_{jl} + B_{ik} \tag{5.12}$$

Its output is a result of a threshold function of this input:

$$v_{ik} = g(u_{ik}) \tag{5.13}$$

The threshold function $g(x)$ is of the form of

$$g(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{5.14}$$

As the solution to equation (5.4) does not change when it is divided by 2, the error function can be halved and mapped to equation (5.11). Expanding equation (5.4), replacing each pixel by the sum of its corresponding nodes as shown by equation (5.10), gives

$$\begin{aligned} E &= \frac{1}{2}(\sum_{s=1}^{N^2}(y_s - \sum_{s=1}^{N^2} h_{si}\hat{x}_i)^2 + \lambda \sum_{s=1}^{N^2}(\sum_{i=1}^{N^2} d_{si}\hat{x}_i)^2) \\ &= \frac{1}{2}\sum_{s=1}^{N^2} y_s^2 - \sum_{s=1}^{N^2}\sum_{i=1}^{N^2}\sum_{k=1}^{M} y_s h_{si} v_{ik} \\ &\quad + \frac{1}{2}\sum_{i=1}^{N^2}\sum_{j=1}^{N^2}\sum_{k=1}^{M}\sum_{l=1}^{M} h_{si}h_{sj}v_{ik}v_{il} \\ &\quad + \frac{1}{2}\lambda\sum_{s=1}^{N^2}\sum_{i=1}^{N^2}\sum_{j=1}^{N^2}\sum_{k=1}^{M}\sum_{l=1}^{M} d_{si}d_{sj}v_{ik}v_{il} \end{aligned} \tag{5.15}$$

where h_{ij} and d_{ij} are the elements of the matrices **H** and **D**, respectively. By comparing the terms in equation (5.11) to the corresponding terms in equation (5.15), the parameters of the Hopfield net are determined as

$$T_{ikjl} = -\sum_{s=1}^{N^2} h_{si}h_{sj} - \lambda\sum_{s=1}^{N^2} d_{si}d_{sj} \tag{5.16}$$

and

$$B_{ik} = \sum_{s=1}^{N^2} y_s h_{si} \tag{5.17}$$

If this Hopfield net is directly implemented, a memory of $O(N^4M^2)$ is required and the time complexity is $O(N^4M^2K)$, where K is the number of iterations. It is difficult to simulate the Hopfield net on a conventional computer even for a 128 × 128 image. However, a practical simulation is possible if the blur function **H** and **D** are local and shift-invariant, for the connection strengths only depend on **H**, the constraint **D** and λ, and it is therefore sufficient to store only the bias term and those nonzero connection weights for each node. Furthermore, a pixel can be represented by a multivalued node, since each binary node which belongs to a given pixel has the same connection weights.

The update rule for each multivalued node x_i can be written as:

$$x_i(t+1) = \begin{cases} x_i(t) + \Delta v_{ik} & \text{if } \Delta E < 0 \\ x_i(t) & \text{if } \Delta E \geq 0 \end{cases} \tag{5.18}$$

where Δv_{ik} is 1,0, or -1 corresponding to the input x_i of positive,zero or negative.

Zhou's algorithm is summarised as follows:

1. Initialise the multivalued nodes to the degraded image.
2. Sequentially visit each node and update it repeatedly until there is no further change. Then move to the next node.
3. At the end of each iteration, check if the energy function has changed; if there is no change, then a restored image is obtained. Otherwise , go back to step 2 for another iteration.

It is obvious that sequential scanning of nodes enforces a preferred direction. In addition, the energy function has to be checked to ensure its convergence since the weight of T_{ikik} may be less than 0. These two properties are undesirable, for it is almost impossible to parallelize the algorithm. A remedy for the energy checking problem is proposed by Paik and Katsaggelos [97]. Their update rule is modified to

$$x_i(t+1) = g(x_i(t) + \Delta x_i) \tag{5.19}$$

where

$$g(v) = \begin{cases} 0 & \text{if } v < 0 \\ v & \text{if } 0 \leq v \leq 255 \\ 255 & \text{if } v > 255 \end{cases} \tag{5.20}$$

and

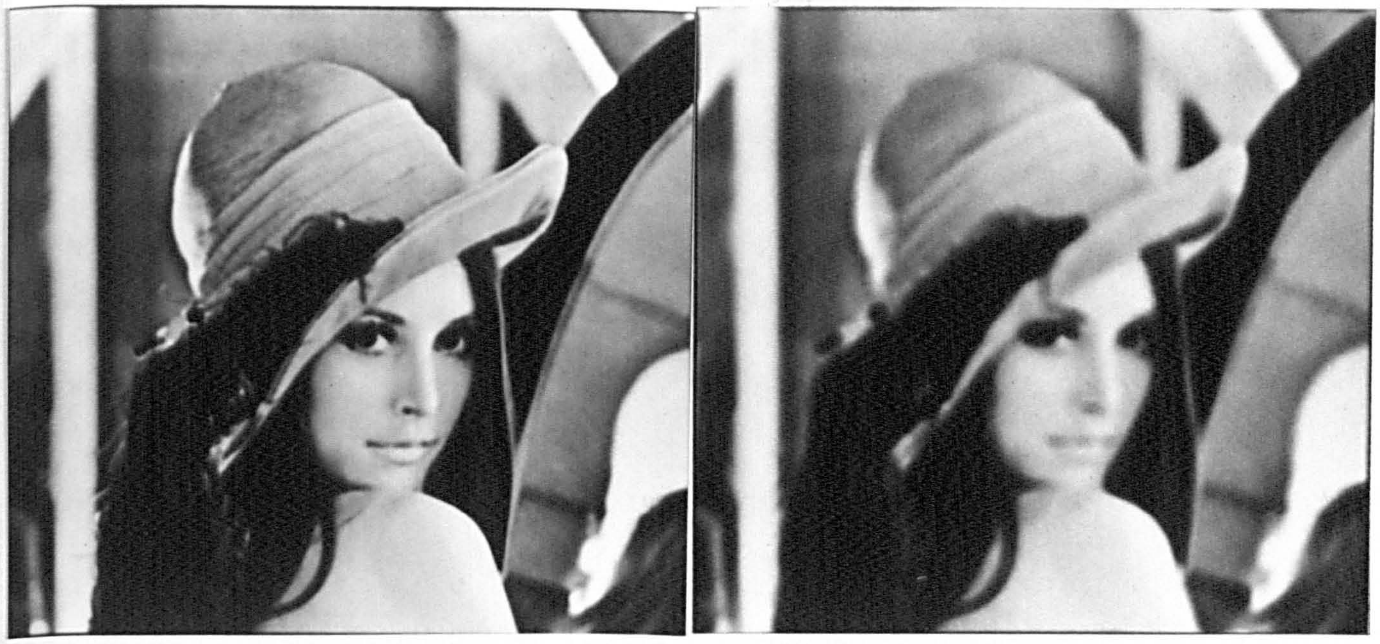
$$\Delta x_i = d_i(u_i) = \begin{cases} -1 & u_i < T_{ii} \\ 0 & T_{ii} \geq u_i \leq -T_{ii} \\ 1 & u_i \geq -T_{ii} \end{cases} \tag{5.21}$$

where

$$u_i = B_i + \sum_{j=1}^{n} T_{ij} x_j(t) \tag{5.22}$$

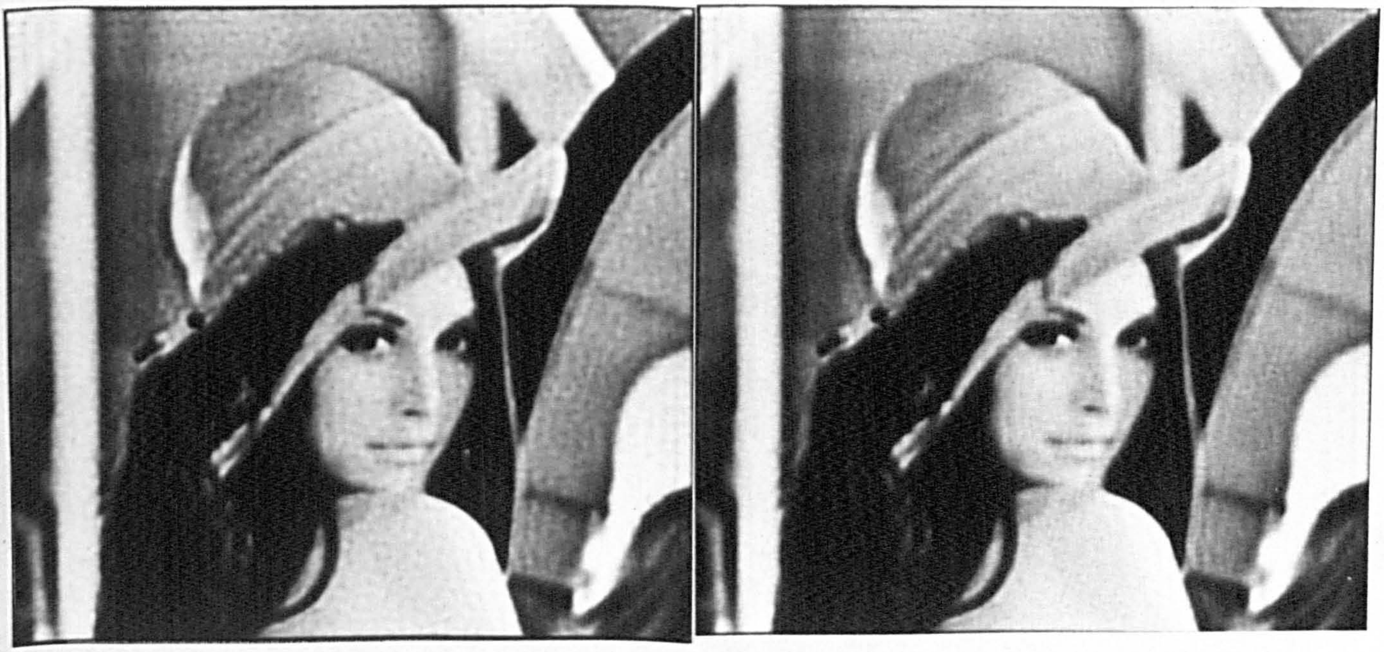
This update rule guarantees energy reduction with negative autoconnections T_{ijij} when the sequential updates are used. A similar result is also demonstrated by Yeh *et al.* [136]. Paik and Katsaggelos also proposed a parallel updating algorithm. However, their parallel scheme is not guaranteed to converge to a local minimum of the energy function of their Hopfield network. The restored image using the network of Zhou *et al.* is known to correspond to a local minimum of the energy function. To find the global minimum, a stochastic decision rule such as simulated annealing can be used [35].

To show how the algorithm works, figure 5.1(a) shows the original 'Lena' image of size 256 × 256 with 256 grey levels and figure 5.1(b) is this image degraded by a 5 × 5 uniform blur function and 30dB additive Gaussian white noise (The variance σ_n^2 of the additive Gaussian white noise is set such that $30dB = 10\log\frac{\sigma_s^2}{\sigma_n^2}dB$, where σ_s^2 is the variance of the original 'Lena' image. Hereinafter, this is used to specify the variances of additive Gaussian white noise used in the test images.) Figure 5.2(a) is the result obtained after 59 iterations. This degraded image is taken as the initial value for restoration. Note the ringing errors manifest at the border areas and edge areas. It is well known that the restoration errors mainly consist of the regularization error and the noise magnification error [73]. Lagendijk pointed out that the regularization error is a function of the source image and this error is related to the local structure of the images, in particular at the edges [73]. The relation between the regularization parameter and restoration errors has been investigated [73][33] and this relation is shown in figure 5.4. Since the regularization error is a monotonic nondecreasing function of the regularization parameter λ and the noise magnification error is a monotonic decreasing one, this implies an optimal choice of λ which can be easily seen from figure 5.5. Zhou's scheme was further tested with the blurred image corrupted



(a) (b)

Figure 5.1: (a) The clean 'Lena' image. (b) The 'Lena' image blurred by a 5 × 5 uniform window and corrupted with 30dB white noise.



(a) (b)

Figure 5.2: (a)The restored image using Zhou's scheme. (b) The restored image using the proposed multiresolution scheme.

with a range of additive white noise. Figure 5.5 shows its performance with a set of different values of regularization parameter λ. As shown, the regularization parameter λ is used to control the trade-off between the two conflicting operations - deblurring and white-noise reduction. When λ is large, the network mainly performs smoothing, so that the SNR of restored images decreases slowly because of the constraint window size. It was observed in the experiment that when the regularization parameter lies in a certain range, the regularization error is mainly caused by the ringing effect, which is due to the bandpass nature of the regularized filter. When the parameter exceeds a certain level, the ringing effect is replaced by blurring. Figure 5.5 shows that the optimal value of the regularization parameter increases while the image is getting more noisy, and finally the regularization parameter is so large that the deblurring is negligible.

As the behaviour of a Hopfield network is very dependent on its connection weights and bias, it is not surprising that the restored images using Zhou's scheme are not far from those restored images obtained from iterative constrained least squares filtering methods [65]. Depending on the connection weights and the initial guess, Zhou's scheme requires several tens of iterations to converge (see Table 5.1). However, the main MSE improvement occurs in the first ten iterations, as shown figure 5.3.

5.3.2 Multiresolution Hopfield Networks for Restoration

In iterative methods, the initial estimate of the solution may affect the convergence behaviour of these methods and possibly improve the restoration results [123]. The Hopfield network proposed by Zhou *et al.* used the degraded image as initial estimate

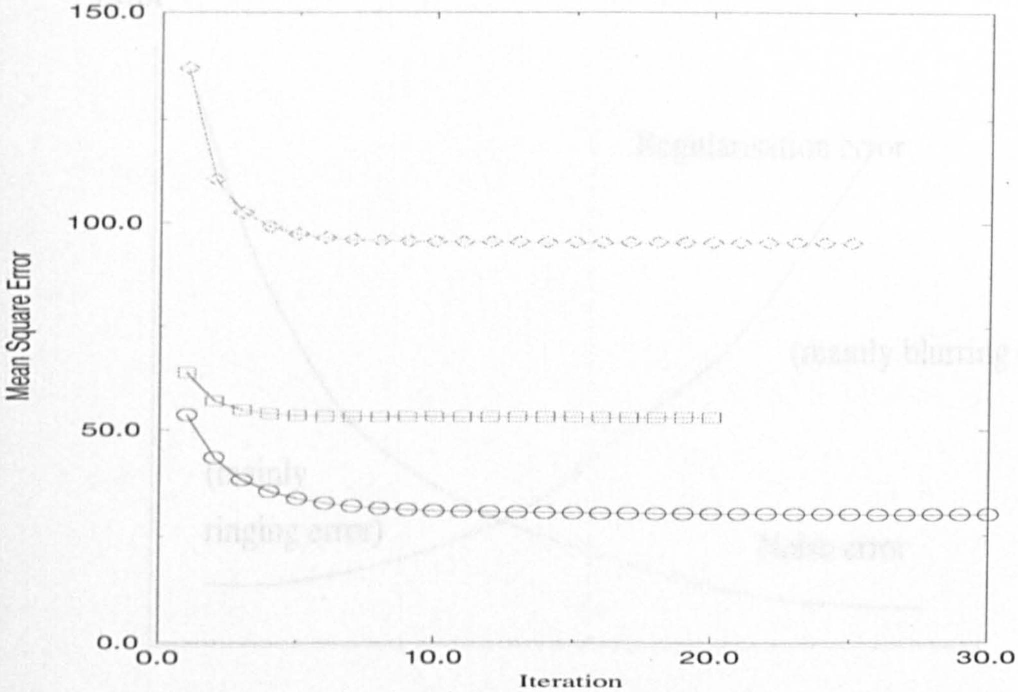


Figure 5.3: The MSE improvement versus the iteration number in Zhou's scheme. $-\circ-$, $-\Box-$ and $-\Diamond-$ are for the 5 × 5 uniform blurred Lena images with 30dB, 20dB and 10dB additive white noise respectively.

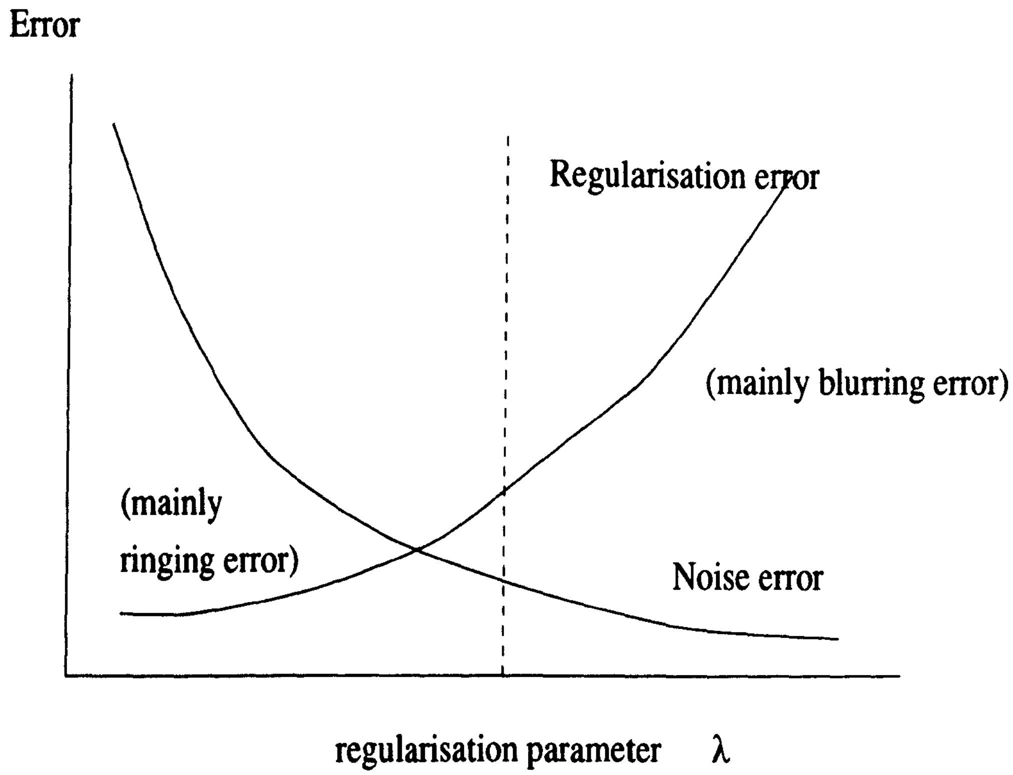


Figure 5.4: The conceptual relation between restoration errors and the regularization parameter

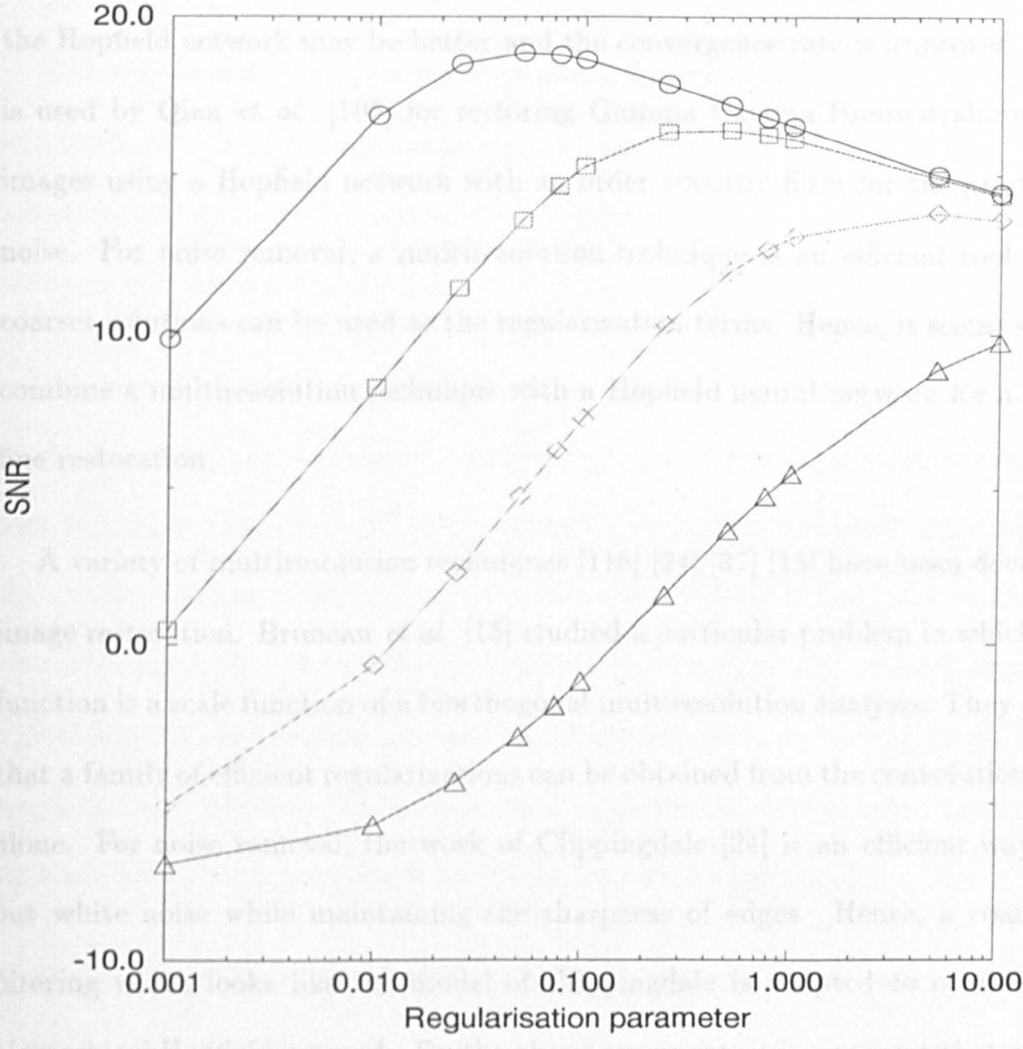


Figure 5.5: The parameter values versus their performance in Zhou's scheme. $-\circ-$, $-\square-$, $-\diamond-$ and $-\triangle-$ are for the 5×5 uniform blurred images with 30dB, 20dB, 10dB and 6dB additive white noise. The degraded images have SNRs 14.1805dB, 13.2856dB, 8.67043dB and 5.46866dB respectively.

of the original image. This is fine when the degradation is mainly caused by blurring and the goal is to sharpen the degraded image . When the image is corrupted down to a low SNR, it is possible to get a better result by using a simple filtering to get a rough estimate as the initial guess of the original image, so that the result obtained by the Hopfield network may be better and the convergence rate is improved. This idea is used by Qian *et al.* [106] for restoring Gamma Camera-Bremsstrahlung medical images using a Hopfield network with an order statistic filter for the pre-filtering of noise. For noise removal, a multiresolution technique is an efficient tool [24]. The coarser solutions can be used as the regularization terms. Hence, it seems sensible to combine a multiresolution technique with a Hopfield neural network for a coarse-to-fine restoration.

A variety of multiresolution techniques [118] [24] [37] [15] have been developed for image restoration. Bruneau *et al.* [15] studied a particular problem in which the blur function is a scale function of a biorthogonal multiresolution analysis. They suggested that a family of efficient regularizations can be obtained from the convolution operator alone. For noise removal, the work of Clippingdale [24] is an efficient way to filter out white noise while maintaining the sharpness of edges. Hence, a coarse-to-fine filtering which looks like the model of Clippingdale is adapted to combine with a hierarchical Hopfield network. By the above argument, it is apparent that combining a multiresolution technique with the Hopfield neural network may also have a possible convergence rate improvement if the initial smoothed estimate is closer to the solution. In addition, it will be a flexible scheme for restoration, in the sense that adaptive filtering is easily implemented.

To use the multiresolution technique, a pyramid of the blurred image is built. Then, a coarse-to-fine restoration is employed. Before the algorithm is outlined, the effects of smoothing and subsampling operations in building a pyramid of the blur image need to be investigated.

It is well known that if the sampling rate is less than the Nyquist frequency, aliasing is introduced [100]. If the decimation factor is 2 in each dimension, a low pass filter whose cutoff frequency is about $\frac{\pi}{2}$ is applied to the image before subsampling to alleviate the aliasing problem [127] .

Apparently, the subsampling by a factor of 2 will reduce the effective window size of the blur by 2 approximately. Therefore, to some extent, the coarser image looks less blurred or sharper, even if there is less information contained. This implies that the coarser solution may be restored more easily as there is less noise and blur in the coarser image. After successive applications of smoothing and subsampling, the blur kernel $b(m,n)$ can be approximated by an impulse $\delta(m,n)$. Using this observation, Lagendijk and Biemond [72] in their work of identification of a large blurring speeded up the identification by using a technique based on a pyramid.

Having built the pyramid of the blurred image, the coarse to fine restoration is summarised as follows:

step 1 Start from level k where the support of the blur can be approximated by a 3×3 window.

step 2 While $(k \leq M)$ do (M is the bottom level)

```
step 3    initialise the Hopfield neural network links, using the level above and the current level to construct the regularization terms.
step 4    While (not converged) run the Hopfield network.
step 5    Propagate the estimate and form the initial estimate for next level.
step 6    k=k+1 ;
```

In step 1, the problem is to determine how many levels to go up so that it is safe to assume the noise is sufficiently reduced. Theoretically, the coarse-to-fine restoration should start from the root level. However, it is not meaningful to start from such a level, for the data at very coarse levels may not be sufficient for estimating statistics reliably. Experiments showed that it should be sufficient to start from a level of size of 64×64 for a blur size of 5×5. The initial estimate for the level is the linear combination of the image at this level and the estimate of the level above. This is a reasonable estimate, for the combination should reduce the effect of noise [23].

The initialisation of the hierarchical Hopfield network is the same as that of Zhou *ét al.*, described in the previous section. However, the smoothness constraint term is replaced by the square sum of the differences between the current level and the estimate of the current level obtained by interpolating the estimate from the level above. This difference can be used to control the value of the regularization parameter for an adaptive restoration, as will be discussed in the next section.

Let $Y(l)$, $\mathbf{H(l)}$ and $X(l)$ be, respectively, the observed data, the blur kernel and the estimate at level l of the blurred image pyramid. The regularization equation is formulated as follows

$$E = \frac{1}{2} \parallel Y(l) - \mathbf{H}(l)X(l) \parallel^2 + \frac{\lambda(l)}{2} \parallel X(l) - \hat{X}(l) \parallel^2 \tag{5.23}$$

where $\hat{X}(l)$ is the propagated estimate for level l from level $l-1$. Note that the constraint parameter $\lambda(l)$ is dependent on the pyramid level. It is not hard to see that the constraint is an approximation of a Laplacian operator used by Burt and Adelson [17].

Expanding equation (5.23) gives

$$\begin{aligned} E(l) &= \frac{1}{2}(\sum_{s=1}^{N^2}(y_s(l) - \sum_{i=1}^{N^2} h_{si}(l)x_i(l)))^2 + \frac{\lambda(l)}{2}\sum_{s=1}^{N^2}(x_i(l) - \hat{x}_i(l))^2 \\ &= \frac{1}{2}\sum_{s=1}^{N^2} y_s(l)^2 - \sum_{s=1}^{N^2}\sum_{i=1}^{N^2} y_s(l)h_{si}(l)x_i(l) + \frac{1}{2}\sum_{s=1}^{N^2}\sum_{i=1}^{N^2}\sum_{j=1}^{N^2} h_{si}(l)h_{sj}(l)x_i(l)x_j(l) \\ &+ \frac{\lambda(l)}{2}(\sum_{s=1}^{N^2} x_s(l)^2 - 2\sum_{s=1}^{N^2} \hat{x}_s(l)x_s(l) + \sum_{s=1}^{N^2}(\hat{x}_s(l))^2) \end{aligned} \tag{5.24}$$

Again, comparing with the energy function of the Hopfield network at level l,

$$E(l) = -\frac{1}{2}\sum_{i,j=1}^{N^2}\sum_{k,l=1}^{M} T_{ikjl}(l)v_{ik}(l)v_{jl}(l) - \sum_{i=1}^{N^2}\sum_{k=1}^{M} B_{ik}(l)v_{ik}(l) \tag{5.25}$$

and disregarding the constant terms $\parallel Y(l) \parallel^2$ and $\parallel \hat{X}(l) \parallel^2$, the connection weights of the hierarchical Hopfield network are given by

$$T_{ikjl}(l) = -\sum_{s=1}^{N^2} h_{si}(l)h_{sj}(l) - \lambda(l)\delta_{ij} \tag{5.26}$$

and its bias term by

$$B_{ik} = \sum_{s=1}^{N^2} y_s(l)h_{si}(l) + \lambda(l)\hat{x}_i(l) \tag{5.27}$$

Note that the bias term in equation (5.27) is reminiscent of the noise filtering operation which is used by Clippingdale[23]. The proposed multiresolution scheme can be approximately considered as using the smoothed image to guide the inverse filtering of the blurred image. The regularization parameter is added to the autoconnections of the network. It is not hard to see that when the regularization is large, the connections will be dominated by the autoconnections. This implies that the solution is approximately the smoothed estimate and the network will converge in several iterations.

After the convergence of the hierarchical Hopfield network at level l, the solution is propagated down and the initial estimate is obtained by the linear combination of the propagated solution and the degraded image at level $l + 1$. The process is repeated until the bottom level of the pyramid is reached.

The multiresolution Hopfield network was tested using the same set of images used to test Zhou's scheme and its performance with a range of values of the regularization parameter is shown in figure 5.6. Comparing with figure 5.5, the performance of the multiresolution scheme is slightly worse than that of Zhou's, in terms of SNR. Note that the performance of the multiresolution scheme is much worse when the value of the regularization parameter is large. This is because when λ is large, the coefficients of the window used for the Hopfield network approximate to an impulse. Hence, the result is close to the interpolation of the starting coarse resolution image. Apparently the multiresolution scheme is more sensitive to the regularization parameter.

Figure 5.2(b) is the image obtained after 56 iterations using the multiresolution Hopfield network. The iteration saving is not impressive, because the degradation is mainly blurring, for which the multiresolution scheme is of limited value.

Visually, it is difficult to tell which is better. Further examining the two results, the multiresolution scheme tends to give a slightly smoother solution. This is demonstrated in figure 5.7. Figure 5.7(a) shows the profiles of the clean Lena, the blurred image with 30dB additive noise and the restored image using Zhou's scheme, while figure 5.7(b) is the same profiles, except that the image restored using the multiresolution scheme replaces that of Zhou's scheme. Note the main difference of the two schemes is at the area about index 40, where the multiresolution Hopfield network gives a smoother result. As the profiles show, errors in edge areas are larger and only adaptive restoration could achieve a better result. Table 5.1 shows the iteration numbers of both Zhou's and the multiresolution scheme. Although in the blurred image with 20dB additive white noise, Zhou's scheme seems to converge faster, the multiresolution scheme has a marginal improvement in convergence rate generally. Note that the number of iterations of the mutiresolution scheme is much smaller when the regularization parameter is large. The performance of the multiresolution scheme can be improved using the difference between the restored and degraded image at coarser

noise/input SNR	Zhou			multiresolution		
	λ	iteration	SNR	λ	iteration	SNR
30dB/14.1805dB	0.05	59	18.8600dB	0.025	56	18.0524dB
20dB/13.2856dB	0.50	20	16.3849dB	0.100	25	15.8278dB
10dB/8.67043dB	4.00	25	13.8377dB	0.350	13	13.3932dB
0dB/0.21889dB	10.0	35	9.7542dB	1.000	4	9.6767dB

Table 5.1: The iteration numbers of Zhou's and the multiresolution schemes applied to a 5×5 blurred image with a range of additive white noise. The regularization parameter λ is about the optimal value.

levels to control the regularization parameter as will be shown in next section.

5.4 Adaptive Restoration Using Hopfield Networks

5.4.1 Adaptive Image Enhancement

In the LSI regularization restoration scheme, the bandpass nature of the regularized inverse filter usually causes the ringing phenomenon [73] and magnifying noise in the smooth regions to be grainy. Indeed, the global smoothness regularization term is not suitable for many images of interest which contain discontinuities. To reduce the ringing phenomenon and the noise in smoothly varying regions, edge areas and smooth areas must be treated differently. Moreover, as the restored image is usually produced for human users, the properties of the HVS should be taken into account in order to get a better subjective quality. The HVS is sensitive to edges, so that a human would prefer an image with sharper edges but a lower MSE to an image with blurred image and a higher MSE.

Various methods have been proposed to retain information which is vital to human

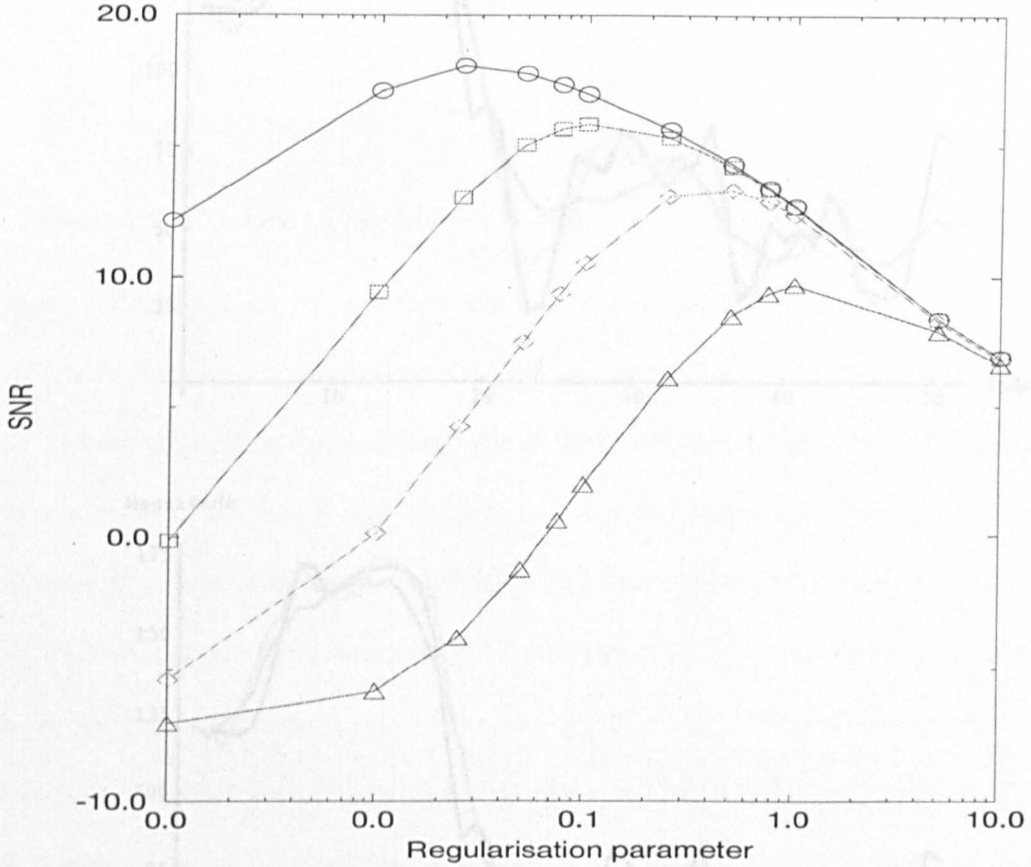


Figure 5.6: The regularization parameter values versus the performance of the multiresolution regularization scheme. -0-, −□−, −◇−, and −△− are for the images with 30dB, 20dB, 10dB, and 6dB additive white noise.

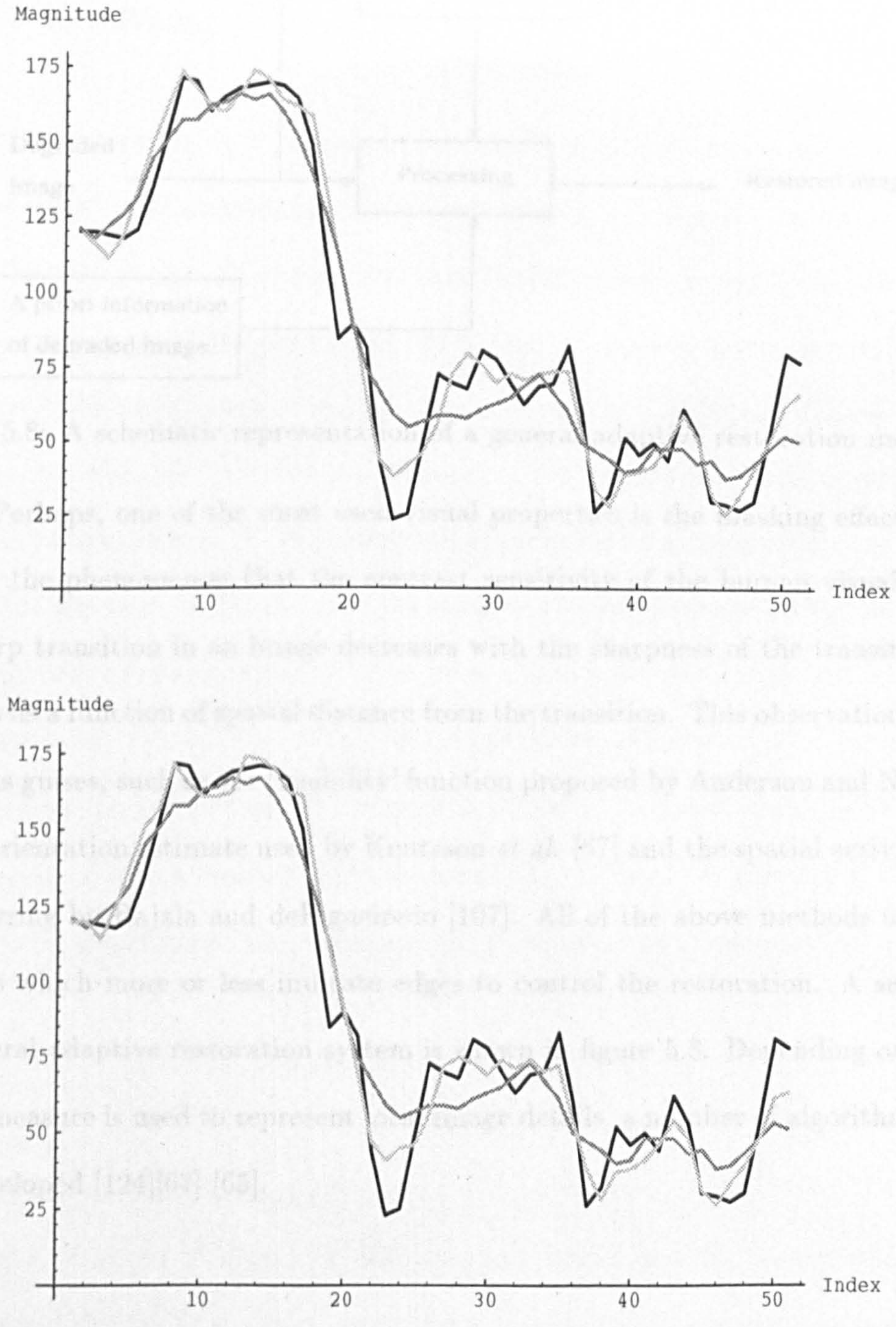


Figure 5.7: (a)The profiles of a horizontal line (130,50)–(130,100) of the original 'Lena' image, the 5x5 uniform blurred with 30dB additive white noise, and the restored image using Zhou's scheme. The darkest line is from the original, the lightest line from the deblurred, and the middle dark line from the blurred. (b) The same profiles with the multiresolution scheme in place of Zhou's scheme.

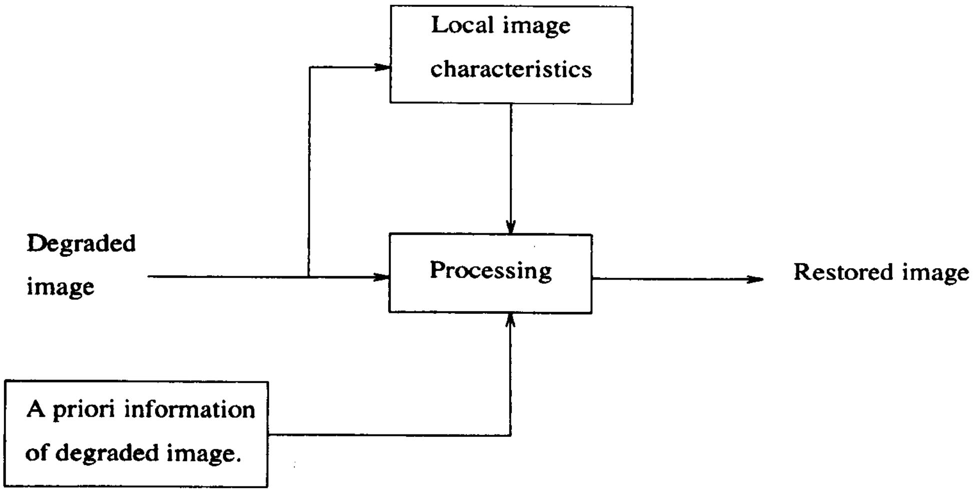


Figure 5.8: A schematic representation of a general adaptive restoration method.

vision. Perhaps, one of the most used visual properties is the masking effect [19][4], which is the phenomenon that the contrast sensitivity of the human visual system at a sharp transition in an image decreases with the sharpness of the transition and increases as a function of spatial distance from the transition. This observation is used in various guises, such as the 'visibility' function proposed by Anderson and Netravali [4], the orientation estimate used by Knutsson *et al.* [67] and the spatial activity used in deblurring by Rajala and deFigueiredo [107]. All of the above methods use local measures which more or less indicate edges to control the restoration. A summary of a general adaptive restoration system is shown in figure 5.8. Depending on which specific measure is used to represent local image details, a number of algorithms have been developed [124][67] [65].

5.4.2 Adaptive Networks for Image Restoration

Adaptive Prefiltering

In adaptive restoration, it is important to identify local characteristics in the blurred image. A simple way to segment the blurred image is to compare the restored image and the blurred image. It is observed that in the smooth regions, the difference is small, but it is large in the edge areas. Therefore, it is possible to segment a blurred image into different regions and use the obtained information for adaptive restoration. Based on this idea, the restored image in the level l of the multiresolution Hopfield network is compared with the degraded images in that level and the difference between the two images is used to control smoothing of the noisy image in the next level. The adaptively smoothed image is then used as the observed image and is restored by the network. Let $t_1(l) < t_2(l)... < t_{n-1}(l)$. The difference image between the restored image and the observed image at level l is propagated down to level $l+1$ and thresholded into n classes. The observed image at level $l+1$ is simply classified into n classes according to the difference image. Each class is smoothed as follows

$$\hat{X}^i(l) = \alpha_i(l)X^i(l) + (1 - \alpha_i(l))\hat{X}^i(l-1) \tag{5.28}$$

where $0 \leq \alpha_1(l) \leq ..\alpha_i(l)..\alpha_{n-1}(l) \leq 1$ are scalar coefficients, and $X^i(l)$ and $X^i(l-1)$ are the sets of pixels of class i in the image of level l and the propagated image $\hat{X}(l-1)$ respectively. This idea of using first a standard CLS algorithm to restore the blurred image for region segmentation for later adaptive restoration is also used in [84].

However, the coarse-to-fine scheme is appealing for computation efficiency compared to Maeda's method[84]. A shortcoming of the multiresolution scheme is that the segmentation is not accurate, but this is always the case, as the blurred image does not provide such information - otherwise the deblurring problem is readily solved.

A range of experiments were conducted to test the performance of the adaptive pre-smoothing method. Two 5 × 5 uniform blurred 'Lena' images with 20dB and 10dB additive noise respectively shown in figure 5.9 are used for the test. The result of applying the adaptive pre-smoothing method to the two degraded 'Lena' images is shown in figure 5.10. The background noise of figure 5.10(a) and (c) is less visible in the smoothly varying areas than that of figure 5.10(b) and (d) obtained by using Zhou's scheme. The edges area of figure 5.10(a) appear to be as good as those in figure 5.10(b). However, as the edges of nose and mouth are not there, the face of the restored 'Lena' seems blurred. In general, the prefiltering scheme does improve the subjective quality and MSE of blurred images in the experiments conducted. Although the adaptive smoothing scheme improves the quality of the restored image, the simple classification of the blurred image into different classes causes a defect on the shoulder area (see figure 5.10(a)). The restored images can be further improved with more accurate edge information provided by a more sophisticated edge detection scheme such as that proposed in chapter 4. The design of an adaptive Hopfield network for restoration can be achieved by imposing various constraints as costs, pixel by pixel or block by block, and mapping the costs to the energy function of the Hopfield network. This can be accomplished by varying the regularization parameters according to the *a priori* knowledge at each pixel as follows

$$E = \sum_i \beta_i (y_i - \sum_k h_{ik} x_k)^2 + \sum_i \lambda_i (\sum_k d_{ik} x_k)^2 \tag{5.29}$$

where β_i and λ_i are scalars which correspond to a local character measurement such as edges.

In equation (5.29), multiplying each pixel by a weighted coefficient is equivalent to a weighted vector norm, which has been widely used for adaptive image restoration [73][120]. Following the notation used by Lagendijk and Biemond[73], the standard regularization for the proposed multiresolution Hopfield network can be written as follows

$$\sum_i \beta_i(l)(y_i(l) - \sum_k h_{ik}(l) x_k(l))^2 = \| Y(l) - \mathbf{H}(l) X(l) \|_s^2$$

$$= (Y(l) - \mathbf{H}(l)X(l))^T \mathbf{S} (Y(l) - \mathbf{H}(l)X(l)) \tag{5.30}$$

$$\sum_i \lambda_i (\sum_k d_{ik} x_k)^2 = \| \mathbf{D}X \|_w^2 = X^T \mathbf{D}^T \mathbf{W} \mathbf{D} X \tag{5.31}$$

where **S** and **W** are diagonal matrices. The diagonal entries of **S** are $S_{ii} = \beta_i(l), 0 \leq \beta_i(l) \leq 1$, the propagated difference between the restored image and the observed image at level $l - 1$. The diagonal entry of **W** is a function of the edge map of the blurred image of level l (for example $\lambda_i = cos(e_i)$ where e_i is the pixel value in the corresponding edge map).

Using the weight vector norm notation, the adaptive CLS criterion used as the

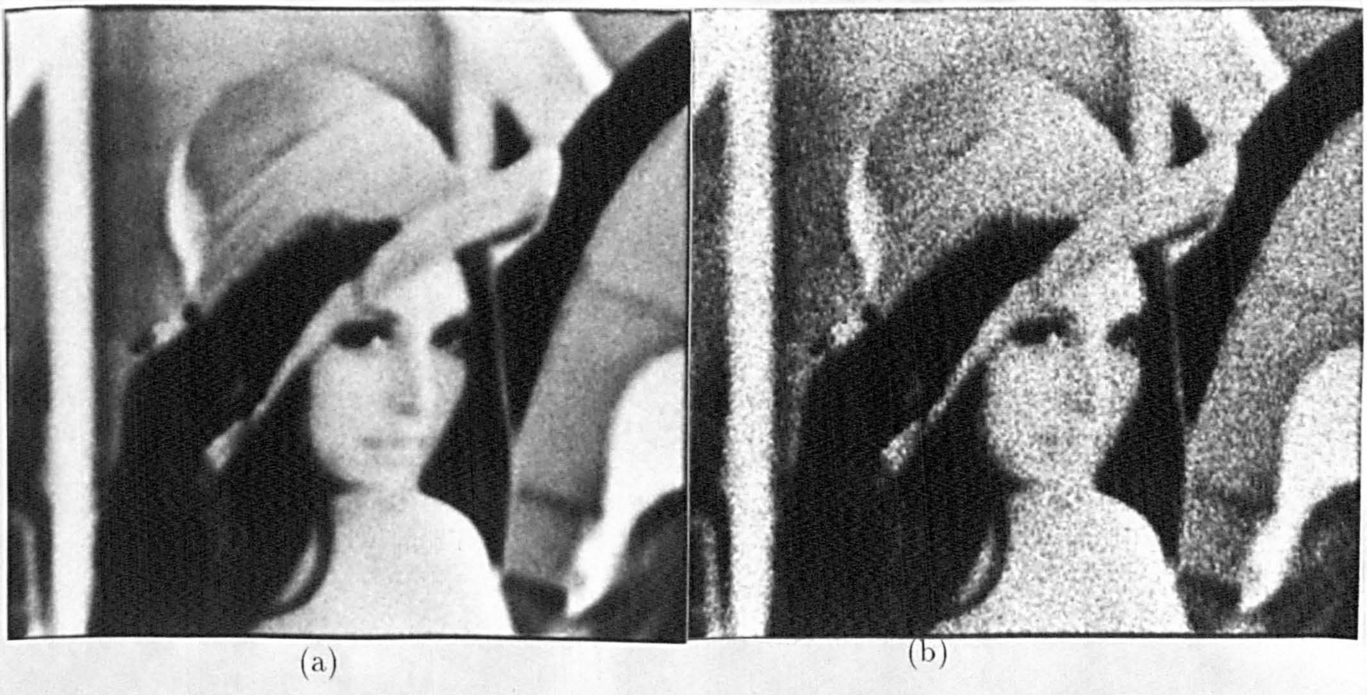
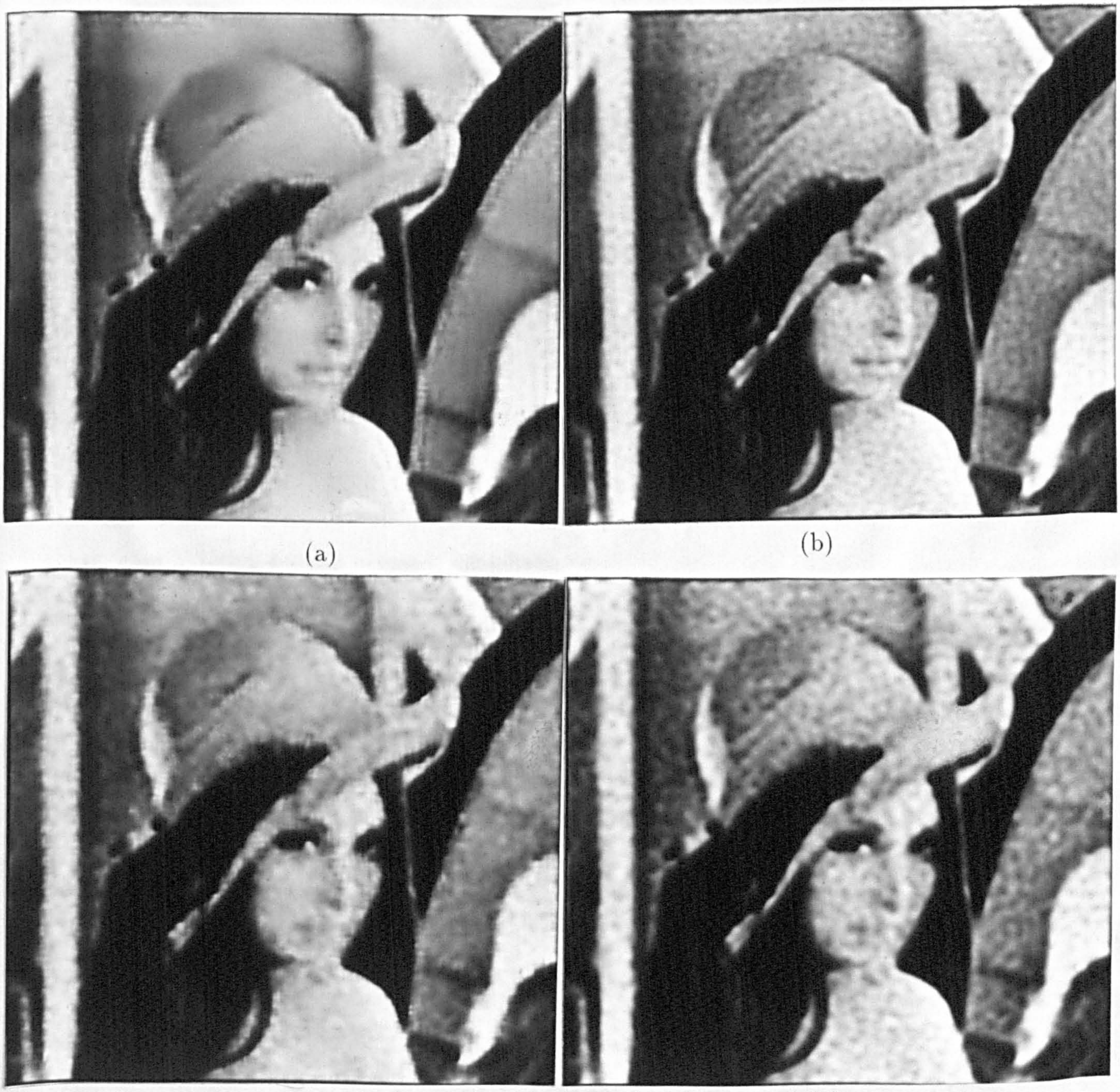


Figure 5.9: Test images (a) The 5 × 5 blurred 'Lena' with 20dB white noise (b)The 5 × 5 blurred 'Lena' with 10dB white noise



(a)

(b)

(c)

(d)

Figure 5.10: (a) The restored 20dB 'Lena' using pre-smoothing. (b) The restored 20dB 'Lena' using Zhou's scheme. (c)The restored 10dB 'Lena' using pre-smoothing. (d) The restored 10dB 'Lena' using Zhou's scheme.

additive noise	Input SNR	Output SNR	λ	Iteration
20dB	13.2856dB	16.4542dB	0.07	34
10dB	8.67043dB	13.9742dB	0.25	16
8dB	7.13479dB	13.3528dB	0.3	13

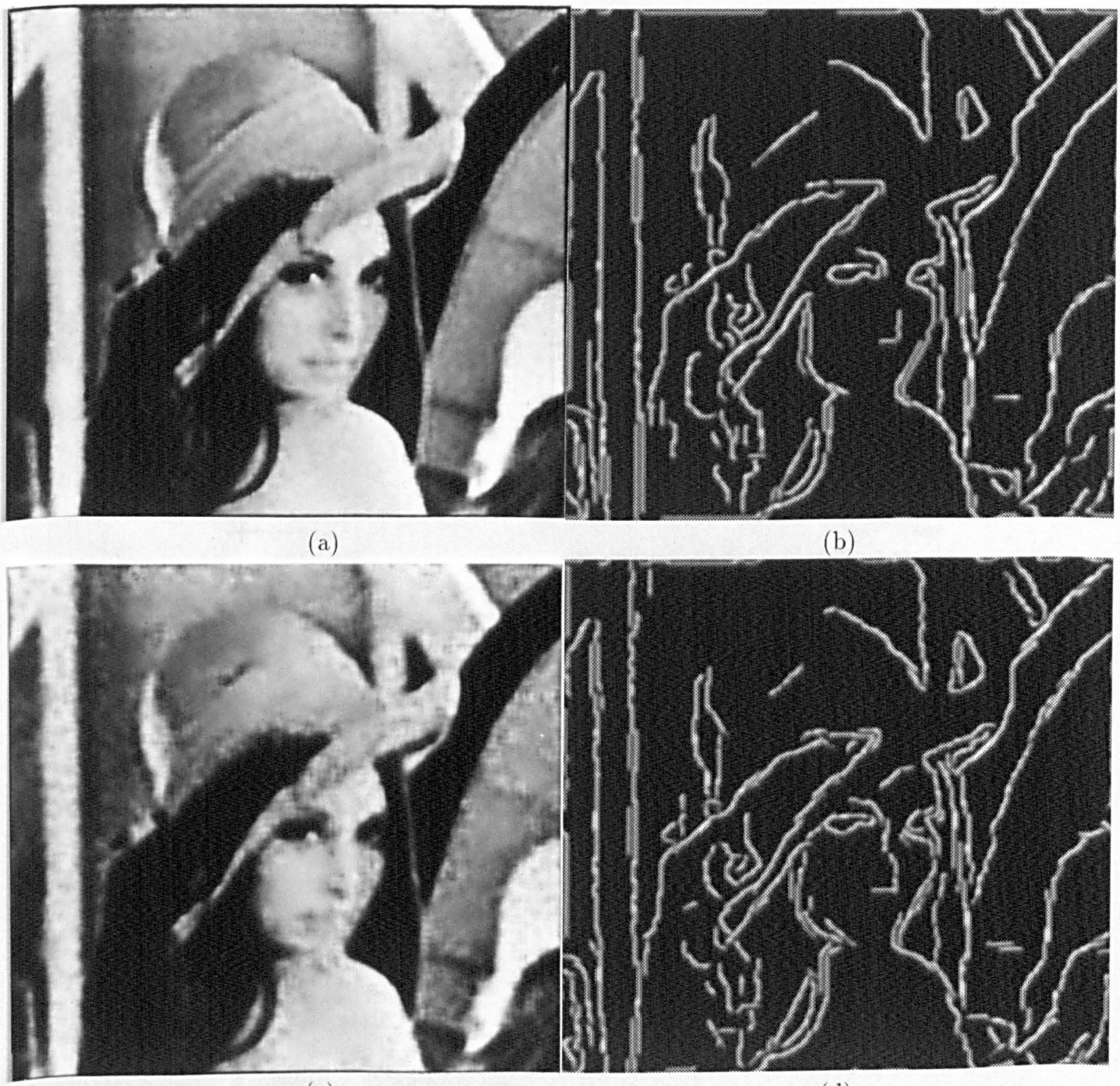
Table 5.2: The SNRs of the degraded 'Lena' images and the restored images using the adaptive multiresolution scheme. The regularization parameter λ is about the optimal value.

energy function for the multiresolution Hopfield network is

$$E(l) = 1/2(||Y(l) - \mathbf{H}(l)X(l)||_s^2 + ||X(l) - \hat{X}(l)||_w^2) \tag{5.32}$$

The weighted CLS criterion is mapped to the energy function of the multiresolution Hopfield network for the network initialisation.

The result of applying the adaptive Hopfield network to the blurred Lena images with 20dB and 10 dB additive white noise are illustrated in figure 5.11(a) and (c). Figure 5.11(b) and (d) are the edge maps used to control the regularization parameter, which are obtained using the edge detector proposed in chapter 4 and then blurred by a 5 × 5 window to reduce the overshooting of restored edges. Also shown in figure 5.12 are horizontal profiles of restored images using Zhou's scheme and the adaptive multiresolution scheme. Table 5.2 summarises the SNR improvement of applying this scheme to the blurred image with a range of different white noise. Apparently, the use of the edge information further improves the performance of the network at the edge area. As the edge numbers constitute a small portion of an image, the improvement of MSE is only marginal in terms of SNR. However, the subjective quality is much improved.



(a) (b)

(c) (d)

Figure 5.11: The restored 'Lena' images using the adaptive scheme. (a) The restored 20dB 'Lena' image. (b) The edge map from the 20dB 'Lena' image. (c) The restored 10dB 'Lena' image. (d) The edge map from the 10dB 'Lena' image.

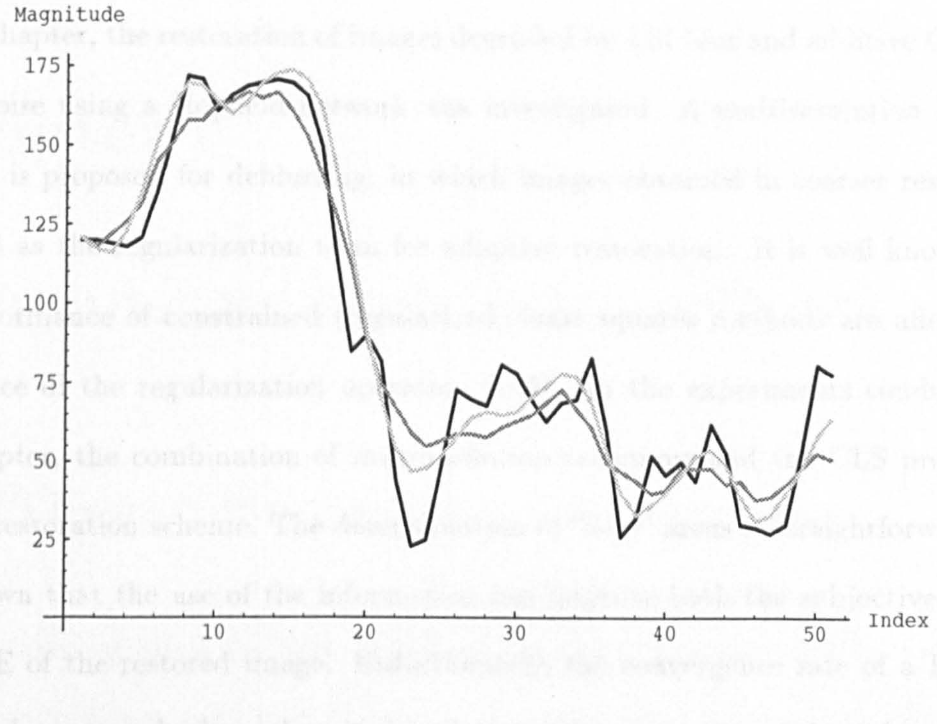


Figure 5.12: (a)The profiles of a horizontal line (130,50)–(130,100) of the original 'Lena' image, the 5x5 uniform blurred with 20 dB additive white noise, and the restored image using Zhou's scheme. The darkest line is from the original, the lightest line from the deblurred, and the middle dark line from the blurred. (b) The same profiles with the adaptive multiresolution scheme in place of Zhou's scheme.

5.5 Discussion

In this chapter, the restoration of images degraded by LSI blur and additive Gaussian white noise using a Hopfield network was investigated. A multiresolution Hopfield network is proposed for deblurring, in which images obtained in coarser resolutions are used as the regularization term for adaptive restoration. It is well known that the performance of constrained (regularized) least squares methods are affected by the choice of the regularization operation [108]. In the experiments conducted in this chapter, the combination of mutiresolution technique and the CLS provides a flexible restoration scheme. The determination of 'busy' areas is straightforward and it is shown that the use of the information can improve both the subjective quality and MSE of the restored image. Unfortunately, the convergence rate of a Hopfield network does strongly depend on its bias input and connection weights. As an image at level l is a lowpassed version of the image at level $l + 1$ and the setting of this Hopfield network is entirely determined by the observed images, it is not surprising that the multiresolution scheme does not improve the convergence rate of the network much at high SNR's. At low SNR's, however, both convergence rate and output SNR are considerably improved by the multiresolution technique. Moreover, the scheme is easily made adaptive by incorporating variation of the regularization parameters based on the edges detected using the method described in Chapter 4. This scheme can give an acceptable estimate of the local activity hence an adaptive restoration scheme with little extra computation.

Chapter 6

Conclusions and Future Work

6.1 Thesis Summary and Contributions

This work investigated a quadratic energy minimization framework for two image processing problems: edge detection and image restoration. Central to this framework are relaxation operations, which can be implemented using a recurrent network. A multiresolution network was proposed to tackle the uncertainty principle problem and the regularization of ill-posed image processing problems.

In chapter 1, the generic properties of images were discussed and it was emphasized that any general image processing system would need to tackle the uncertainty problem, scale and distortion, and to apply domain knowledge for 'regularizing' ill-posedness. An interesting approach to incorporate priori knowledge is to use an energy function to specify relations between structures and select a solution that gives the minimum of the energy function [104] [113]. This principle of regularization theory was adopted to solve ill-posed problems.

Since the HVS outperforms many conventional algorithms, it was argued that the methods used by the HVS may provide insights into developing a robust computer vision system. Following this argument, models, though still crude, have been developed for a better understanding of the HVS [80] [81] and the use of ANNs for image processing is fruitful and expanding rapidly. A survey of ANNs and their application to image processing was given and the advantages of such an approach in contrast to conventional algorithms were discussed.

Chapter 1 was concluded with a need to unify a set of requirements into a model for computer vision and image processing, which include

1. Tackling the uncertainty problem: How to trade-off localisation and noise immunity in different resolutions.
2. The use of *a priori* knowledge: How to incorporate prior knowledge.
3. Global/local processing: How to compute local features and incorporate them into a globally coherent description.
4. Flexibility : The developed model should be readily extensible.

In chapter 2, various image models were reviewed, in particular a class of multiresolution models was emphasized, in view of the need to deal with the uncertainty problem in image processing and vision [129]. A multiresolution algorithm combined with an iterative scheme was proposed for edge detection and image restoration.

An image model provides a framework within which various image processing techniques can be designed and analyzed. Stochastic modelling of images treats the spatial coordinates of an image as time-like indices so that models for 1-D signals can be borrowed by introducing raster scanning. 1-D stochastic models, however, lack the facility to take into account the spatial structure inherent in a two dimensional image. In 2-D stochastic modelling of images, the local spatial structures of the image can be modelled using a Markov random field. Alternatively, Gaussian Markov random fields can be represented as the outputs of linear systems described by difference equations, whose inputs are random fields with known or desired properties. Although these 2-D models do take the spatial features into account, they lack a proper method for describing features in different scales. Moreover, the uncertainty problem naturally leads to a multiresolution representation, in which small features are modelled at higher resolutions while large features are modelled at lower resolutions. An important feature of multiresolution representations is their ability to render features which span a range of scales and which are essential for perception. As features may appear at different locations at different resolutions and they may also disappear at several resolutions– a method is needed to combine information obtained in different resolutions, if multiresolution representations are used [82]. This is the so called *scale consistency* problem [132]. To tackle these problems, a class of stochastic multiresolution models which were based on representations such as pyramids and wavelets were recently proposed [24] [21]. Such modelling is able to render a variety of images adequately and provides an efficient estimation procedure for applications such as estimation[23], segmentation[10], optical flow [83] and feature extraction[18]. This class of stochastic multiresolution models is causal in the scale axis, which entails a coarse-to-fine estimate for restoration and feature tracking for edge detection, but they are noncausal in the spatial coordinates.

The generalised multiresolution model on a pyramidal representation used in [10] was adapted and an algorithm based on this representation was outlined for edge detection and adaptive image restoration. The use of relaxation in a multiresolution model improves long-range characteristics of the model. The information in coarser resolutions is used to *regularize* the solution in successive finer resolutions until the highest resolution is reached. Vertical propagation is also used, with the aim of emphasizing consistent features and reducing uncorrelated noise. For each resolution, a relaxation scheme based on an energy minimization framework is used to update the confidences of the initial estimate and decisions. Information flows vertically and horizontally in the model, which is reminiscent of the information processing in the HVS[85].

Having outlined the model, the implementation is equally important because different architectures will affect the implementation [88]. The HVS, which is an appropriate computational model for vision, and the most successful implementation ever known to man, was examined in Chapter 3, with the hope that its properties would bring insights to the problem of computer image processing and vision. The HVS's sensor is the retina, which can adapt to a wide range of lighting conditions using temporal mechanisms [53] . In addition to adaptation, study of optical illusions has shown that the retina reduces the bandwidth of visual data and extracts only the essential features of an image. Thus, Marr suggested that a primal sketch is used by the HVS [88]. Although the retina contains a blind spot, psychophysical experiments show that a line across the blind spot will appear unbroken. The implication is that the visual system uses *a priori* knowledge[25]. In the visual cortex, structures become more complex and the degree of abstraction increases. There are simple cells which respond to edges and lines [55] and end-stopped cells which were suggested to implement end-point detection [29]. Moreover, study of patients with partial visual cortex damage shows that visual cortex is divided into several task-specific regions [138]. This suggests that visual perception is the result of several different processes.

To encapsulate these features of HVS, a hierarchical Hopfield network was proposed. This network implements the image model proposed in chapter 2, in which each resolution corresponds to a Hopfield network. The solution for each resolution is obtained after the Hopfield network converges. This work shows that the multiresolution model can easily incorporate smoothness constraints to regularize the solution space. In Chapter 3, it was concluded that the benefits of using a multiresolution Hopfield network are fourfold: (1) to speed up its convergence rate, (2) using lateral interaction to reflect the intrinsic local structures of images, (3) smoothness constraints and (4) noise immunity.

In chapter 4, an edge detection scheme was detailed, based on the multiresolution model outlined in chapter 2 and was implemented using the proposed hierarchical Hopfield neural network. There are two steps in the edge detection algorithm. The first step in the edge detection scheme is to construct the orientation vector field pyramid of an input image. The second step is to construct a compatibility function for the interaction of two neighbouring pixels and an energy function based on the compatibility function as the energy function of the Hopfield network. The compatibility function is defined to reflect the fact that when two pixels are in the same shape, the measure is high; otherwise it should be low. This was achieved using local orientation information to group edge pixels. In other words the compatibility function reinforces continuity and smoothness of edges. To ensure that the resultant edge is as thin as possible, the compatibility measure also includes lateral inhibition so that the winner of a competition among pixels across the direction of the edge is selected as the edge. The compatibility measure also takes distance between two pixels into account using a lowpass-like weighting function. Finally, the magnitude of the orientation vector also plays an important role: the more certain is the orientation, the stronger should be the link. In short, the compatibility measure between two edge components is separable into a function parallel to the edge, expressing continuity and proximity and one perpendicular to the edge, expressing lateral inhibition.

After the setting of the Hopfield neural network for level l , the edge map of level l is obtained when the Hopfield neural network converges. Using the resulting edge map, the orientation estimate is refined and it is propagated down to level $l+1$ to refine the estimate by a linear combination of estimates in the two levels. The new estimate of orientation for level $l+1$ is then used to set the energy function of the hierarchical Hopfield network for this level and the edge map for level l is used as bias for the Hopfield neural network. Again the edge map for level l is obtained, once the Hopfield neural network converges. The coarse-to-fine process proceeds recursively until the bottom of the pyramid is reached.

The application of the Hopfield network to several test images showed that this simple coarse-to-fine process with lateral interaction is robust and gives results of at least as good a quality as those from a Canny hysteretic edge detector with thinning.

Compared with the results of Hérauld and Horaud [45], it seems that the results of this work are more robust, for the coarse-to-fine process provides an effective immunity facility to noise, although there is no quantitative comparison here. Moreover, the Hopfield network requires no parameter tuning and is amenable to parallel implementation.

In chapter 5, the proposed hierarchical Hopfield neural network is used for image restoration. It is shown that in the hierarchical Hopfield neural network approach, the 'coarser' resolution solutions can be used in a regularization term and they can be also used to segment the images in the higher resolutions into regions of different activity so that a coarse-to-fine adaptive noise filtering is possible [23]. To further improve the performance of the multiresolution restoration scheme, an adaptive restoration scheme which uses the edge map obtained by applying the Hopfield network detailed in chapter 4 was proposed.

The first task in using a Hopfield network for image restoration is to construct a suitable energy function for the Hopfield network. This work adopted the methods of Zhou *et al.* [141] and used a quadratic regularization function as the energy function of a Hopfield network. The convergence rate of Zhou's scheme was shown to be somewhat improved. As restoration of blur and additive noise is considered, the effect of of smoothing and subsampling operations in building a pyramid of the blur image was discussed.

In the hierarchical Hopfield network, the difference between the interpolation of the solution in the level above and the solution of the current level approximates a

Laplacian operator [17], and a linear combination between the coarser solution and the current data as the initial estimate may be closer to the solution in the sense that noise is reduced, hence reducing the number of iterations. An experiment was also conducted to test the performance of the hierarchical Hopfield network. It was shown that the convergence rate is improved if noise is the main source of corruption. The reason is that the multiresolution model is effective in noise filtering.

The phenomenon that there is less noise and less blur in the coarser resolutions was exploited for adaptive filtering. A simple way to segment the blurred image was used by comparing the restored image and the blurred image. It was observed that in the smooth regions, the difference was small, but it is large in the edge areas. Therefore, a scheme which roughly segments a blurred image into different regions and uses the information for an adaptive noise suppression was proposed. The restored images can be further improved with more accurate edge information provided by a more sophisticated edge detection scheme, such as that proposed in chapter 4.

The design of an adaptive Hopfield network for restoration can be achieved by imposing various constraints as costs, pixel by pixel or block by block, and mapping the costs to the energy function of the Hopfield network. This was accomplished by varying the regularization parameters according to the estimated properties at each pixel. This chapter was concluded with the results of this restoration scheme.

6.2 Limitations and Future Work

The results presented in this work were based on a multiresolution optimization framework, implemented by a hierarchical Hopfield neural network. Although they show the effectiveness of this approach, a number of issues about the model and algorithms need to be considered further, so that its potential and limits can be fully realised.

The proposed model consists of two processes: a coarse-to-fine process and a lateral iterative optimization process for each different task. The model may be further improved to allow the interaction between different task processes: the edge detection process and the restoration process. Although edges were used for adaptive restoration and when there is no blur in the input images, the edge detection process is effective. However, as shown in chapter 5, when the input image is distorted by blur as well as noise, it seems that the deblurred image can be used as input for edge detection so that a more accurate result can be obtained. In short, the information used for the lateral iteration is mainly of one type. There is a feeling that there should be a co-operation between lateral processes which extract or estimate different characteristics of images, such as edge, grey level and texture (eg. Bhalerao[10]). Figure 6.1 shows the idea.

A number of algorithmic and implementation considerations also are indicated by the analysis of the results. First, in the edge detection scheme, they are:

1. Estimation of coefficients for cross level averages: the coefficients of the linear multiresolution model for cross level refinement for level l and $l + 1$ is a function

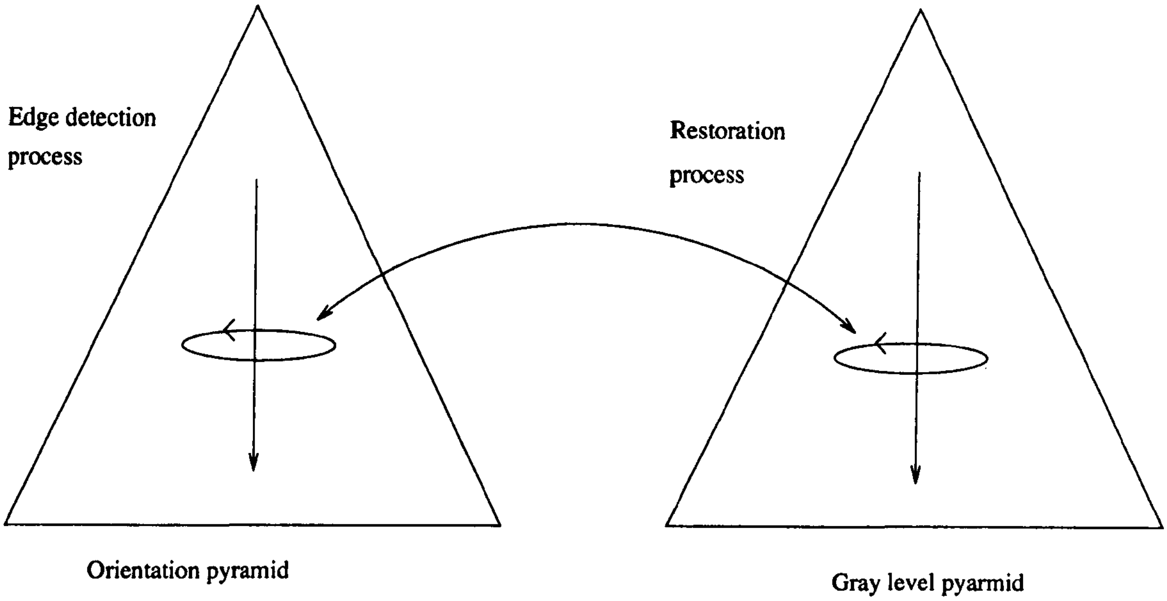


Figure 6.1: A possible refinement of the edge and restoration model

of the edge map obtained at level l and signal to noise ratio of both levels. This setting will suppress noise as well as new weak edges at level $l+1$. Consequently, it is possible that the edge scheme will miss weak edges. (cf. figure 4.4 and figure 4.11). A possible remedy may take orientation structure at level $l+1$ for determination of the cross-level combination coefficients. This remains to be investigated.

2. Image properties used for edge determination: the edge detection scheme proposed in this work utilises mainly a 5×5 support of orientation information to detect edges. The network provides an iterative local computation for edge grouping. It is apparent that such an iterative scheme may slowly integrate local clues to form a consistent global edge map and the multiresolution may help to speed up this process. However, the scheme has some defects in registering corners and junctions properly because it uses only correlation, while corners

and junctions require higher order relations to be determined. To use the same Hopfield network structure, such information can be fed in using the bias term. Another thought is to use a more complex network which exploits higher order spatial relations, for example [93]. Another defect of this scheme is that it may give spurious (ie. parallel lines) or thick edges for blurred edges, depending on the width of the window used. This is, however, a defect of the underlying sequential scan Hopfield network algorithm.

For the adaptive restoration scheme, the segmentation and the cross level combination coefficients are chosen heuristically. There is a need to make both of these choices more rigorous. However, its attraction is its simplicity and to some extent, it is robust.

6.3 Concluding remarks

This work has tried to unify into a framework a number of different ideas which have been used in various image processing problems and computer vision, in particular, edge detection and image restoration. Multiresolution analysis and regularization theory are the theoretical tools used. The results presented show that these techniques are an effective and robust way of using neural networks for solving the problems of image processing and low-level vision.

Through evolution, nature has come up with a solution to vision problems: the HVS. It is, thus sensible to borrow some of the principles used in the HVS – the most powerful vision system known. Not surprisingly, study of the HVS has already had an impact on algorithm development for computer vision and image processing. An important feature of neural networks is that they process information iteratively, which together with multiresolution analysis and regularization theory inspired the hierarchical Hopfield network for edge detection. What is clear is that much work is still to be done in the area of how to integrate consistent and contradictory clues to give a globally coherent solution.

Of course, there are many aspects of neural networks that need to be investigated and put into use in computer vision. Among them, the learning capacity of neural networks may be a key factor in the various remarkable abilities of the HVS. No doubt learned knowledge can be used as a priori knowledge to regularize the ill-posed early vision processes in an adaptive vision system. It is hoped that in the future, a learning scheme can be incorporated to give a more robust system.

Appendix A

Conference Paper

This paper has been presented on the eighth IEEE workshop on Image and Multidimensional Signal Processing[135].

Orientation-directed edge detection using a Hopfield neural network

H. C. Yang, R. Wilson

University of Warwick,UK

Abstract

A Hopfield neural network can solve image processing tasks which can be posed as optimization problems(eg. [3]). In this work, edge detection is formulated as an optimization problem. A multiresolution image model, combined with local interaction in each level of resolution, is used for the edge detection problem. The model is implemented by a hierarchical Hopfield neural network. We show that edge detection can be done well by the coarse-to-fine Hopfield network whose connections are orientation-tuned, which has strong motivation from mammalian visual systems[1].

1 Method

Combined with a multiresolution model, a hirerarchical Hopfield neural network is shown to be efficient for edge detection. the edge detection problem is formulated as an optimization problem; Thus a cost function is defined and used as the energy function of the Hopfield neural network. The algorithm consists of two stages. In the first stage a grey level pyramid of an input image is built. An orientation pyramid is then built of the image by applying a pair of highpass filters to the grey level pyramid. These filters are specially designed to reduce orientation bias. The orientation field is a double angled one, which eliminates ambiguity in the gradient representation of orientation[2].

The second stage can be best summarized by the following pseudo code;

```
1   from level k in the orientation pyramid. (* the image plane is level 0*)
2   while ($k \geq 0$){
3       initialize the Hopfield neural net using the orientation estimate;
4       while not converged{
            run Hopfield neural net, using level k+1 as bias}.
5       refine the orientation estimates using the egde map obtained from the network.
6       estimate signal to noise ratio at level k;
7       if ($k \geq 1$){
8           estimate signal to noise at level k-1;
9           linearly combine the initial estimate at level k-1 and the propagated
            estimate from k to give the refined orientation estimate;}
10      k = k-1.}
```

Note each pixel is represented by a unit in the Hopfield neural network. In step 3, a measure of compatibility between two pixels is defined as follows: Let i be the i-th neuron in the Hopfield network and $\vec{v_i}$ be the double-angled orientation estimate at pixel (x_i, y_i); the compatibility between two pixels i and j will be measured as follows

$$c_{ij} = k(\sigma^2 - b^2)exp(-b^2/2\sigma^2)cos(2\pi a/(2w-1)) \tag{1}$$

where k is a constant, w is the window size, σ is a function of w and

$$a = \gamma cos\theta$$

$$b = \gamma sin\theta$$

where

$$\theta = \frac{1}{2}arg(\vec{v_i} + \vec{v_j}) - arg((x_i - x_j, y_i - y_j))$$

$$\gamma = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

In other words, the compatibility between two edge components is separable into a function parallel to the edge, expressing continuity and one perpendicular to-the edge, expressing lateral inhibition. After the Hopfield network converges, those pixels whose corresponding units are on, are chosen as edge locations.

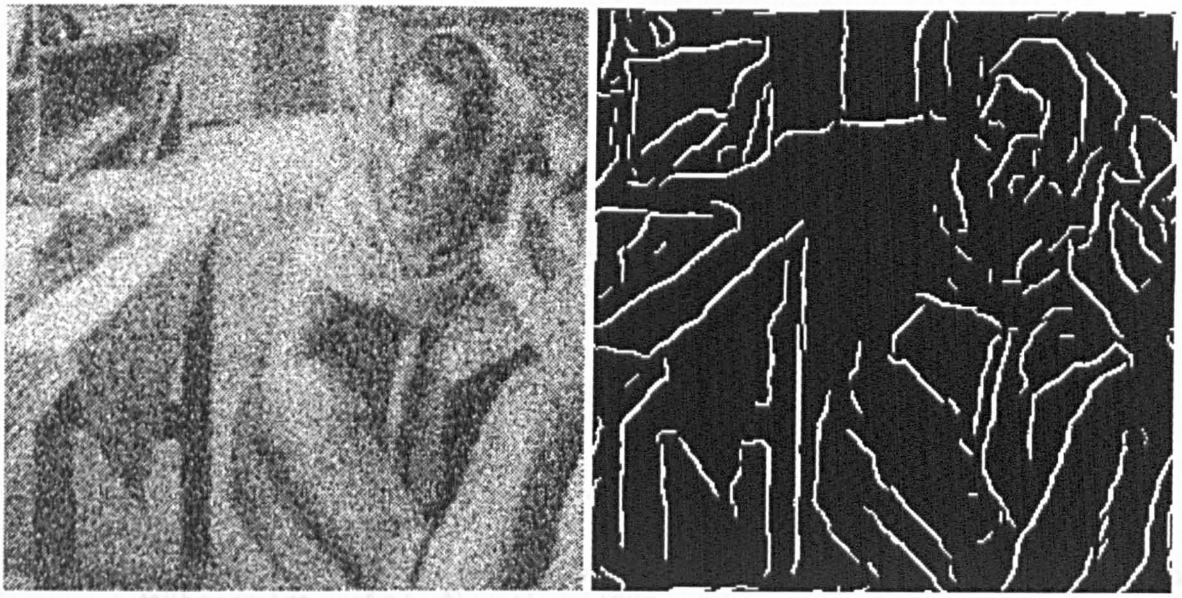


Figure 1: (a) a 256x256 test image of 0db SNR (produced by an HP laserprinter, dithered) (b) its edge map obtained by the Hopfield neural network.

The image in this level is, thus divided into edge areas and noisy background. The orientation estimates are refined in such a way that these estimates in noisy areas are averaged isotropically while the estimates in the edge areas are averaged in the direction of their orientations(step 5). Only those orientation estimates in the edge areas are propagated down. The SNR in this level is estimated as the ratio of the variance of the orientation vectors in edge areas and that of noisy areas. To get a better estimate of SNR at the next level, the edge map is propagated and the image is accordingly divided into 'noise' areas and 'signal' areas. The coefficient for the linear combination of the initial orientation estimate and the propagated estimate is simply the ratio of $\mathrm{SNR}_k/(\mathrm{SNR}_{k-1}+\mathrm{SNR}_k)$. In short, the noisier the initial estimate is, the more comes from the propagated estimate from which noise has been filtered. The above steps are repeated down to the image plane.

2 Results and conclusions

Figure 1a, shows a test image of size 256 x 256. White noise is added to give an SNR of 0db. Figure 1b. is the edge map obtained when the Hopfield neural network converged. The window size used is 7 pixels. Typically the network will converge in about 8–10 iterations. The 7 x 7 window used is, more or less, a template of a short line. The weighted sum of input received by each neuron in the Hopfield network thus is viewed as a measure of certainty of a line being present. In addition, the iteration of the network provides line refinement; hence curves can be detected. The results represent a significant improvement over techniques employing simpler enhancement strategies(eg.[2]).

References

[1] D. H. Hubel and T. N. Wiesel. Brain mechanisms of vision. *Scientific American*, pages 130–144, 1979.

[2] R. Wilson, S. Clippingdale, and A.H. Bhalerao. Robust estimation of local orientations in images using a multiresolution approach. In *SPIE, Visual Communications and Image Processing*, pages 1393–1403, Lausanne, Switzerland, 1990.

[3] Y-T Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins. Image restoration using a neural network. *IEEE Tran. ASSP*, 36(7), July 1988.

Bibliography

[1] E. Aarts and J. Korst. *Simulated Annealing and Boltzmann Machines.* John Wiley and Sons Ltd., New York, 1989.

[2] K. Abend, T. Harley, and N. Kanal. Classification of binary random patterns. *IEEE Trans. Information Theory*, 11:538–544, 1965.

[3] A. A. Amini, T. E. Weymouth, and R. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Trans. PAMI.*, 12(9):855–867, 1990.

[4] G. L. Anderson and A. N. Netravali. Image restoration based on a subjective criterion. *IEEE Trans. Sys, Man Cyber.*, 6:845–853, 1976.

[5] M. Antonini, P. Mathieu M. Barlaud, and J. C. Feauveau. Multiscale image coding using the Kohonen neural network. *SPIE, Visual Communications and Image Processing*, 1360:14–26, 1990.

[6] M. Basseville, A. Benvenite, K. C. Chou, R. Nikoukhah S. A. Golden, and A. S. Willsky. Modeling and estimation of multiresolution stochastic processes. *IEEE Trans. Information Theory*, 38(2):766–784, March 1992.

[7] R. Battiti. *Multiscale methods, Parallel Computation and Neural Networks for Real-time Computer Vision.* PhD thesis, California Institute of Technology, 1990.

[8] L. Bedini and A. Tonazzini. Image restoration preserving discontinuities: the Bayesian approach and neural network. *Image and Vision Computation*, 10(2):108–118, March 1992.

[9] J. Besag. On the statistical analysis of dirty pictures. *J. R. Statist. Soc. B*, 48(3):259-302, May 1986.

[10] A. H. Bhalerao. *Multiresolution Image Segmentation.* PhD thesis, The University of Warwick, UK, 1991.

[11] J. Biemond, R. L. Reginald, and R. M. Mersereau. Iterative methods for image deblurring. *Proceeding of The IEEE*, 19(6), Nov/Dec 1989.

[12] M. Boldt, R. Weiss, and E. Riseman. Token-based extraction of straight lines. *IEEE Trans. Sys, Man Cyber.*, 78(5), May 1990.

[13] J. Bruck. On the convergence properties of the Hopfield model. *Proceeding of The IEEE*, 78(10), October 1990.

[14] J. Bruck and J. W. Goodman. A generalized convergence theorem for neural networks. *IEEE Trans. Information Theory*, 34(5), September 1988.

[15] J. M. Bruneau and P. Mathieu M. Barlaud. Image restoration using biorthogonal wavelet transform. In *SPIE, Visual Communications and Image Processing*, pages 1404–1414, Lausanne, Switzerland, 1990.

[16] P. J. Burt. Smart sensing within a pyramid vision machine. *Proceeding of The IEEE*, 76(8), August 1988.

[17] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Trans. Comp.*, COM-31:532-540, 1983.

[18] A. Calway. *The Multiresolution Fourier Transform: A general Purpose Tool for Image Analysis.* PhD thesis, The University of Warwick, UK, 1989.

[19] J. C. Candy and R. H. Bosworth. Methods for designing differential quantizers based on subjective evaluations of edge busyness. *Bell System Tech. Journal*, 51:1495-1516, 1972.

[20] J. C. Canny. A computational approach to edge detection. *IEEE Trans. PAMI.*, 8(6):679-698, 1986.

[21] K. C. Chou, A. S. Willsky, A. Benveniste, and M. Basseville. Recursive and iterative estimation algorithms for multiresolution stochastic processes. In *Proc. 28th IEEE Conf. Decision Contr.*, Tampa, FL, 1989.

[22] A. Cichocki and R. Unbehauen. *Neural Networks for Optimization and Signal Processing.* John Wiley and Sons Ltd, England, 1993.

[23] S. C. Clippingdale. *Multiresolution Image Modelling and Estimation.* PhD thesis, The University of Warwick, UK, 1988.

[24] S. C. Clippingdale and R. Wilson. Quad-Tree Image Estimation: A New Image Model and its Application to MMSE Image Restoration. In *Proc. 5th Scandinavian Conf. Image Analysis*, pages 699–706, Stockholm, Sweden, 1987.

[25] F. Crick and C. Koch. The problem of consciousness. *Scientific American*, 267(3):111-117, September 1992.

[26] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Commun. Pure Appl. Math.,*, 91:909–996, 1988.

[27] J. G. Daugman. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 36:1169–1179, 1988.

[28] A. R. Davies and R. Wilson. Curve and corner extraction using the multiresolution fourier transform. In *Proceedings of the 4th IEE conference on Image Processing and its Applications*, pages 282–285, Maastricht, the Netherlands, 1992.

[29] A. Dobbins, S. W. Zucker, and M. S. Cynader. Endstopped neurons in the visual cortex as a substrate for calculating curvature. *Nature*, 329:438–441, 1987.

[30] P. C. Dodwell. The Lie transformation group model of visual perception. *Perception and Psychophysics*, 34:1–17, 1983.

[31] J. C. Feauveau and E. Viennet. Multiresolution segmentation by neural networks. *International Neural Network Conference, 1990, Paris, France*, pages 149-152, 1990.

[32] D. Gabor. Theory of Communications. *Proc. IEE*, 93(26):429–441, November 1946.

[33] N. P. Galatsanos and A. K. Katsaggelos. Methods for choosing the regularisation parameter and estimating the noise variance in image restoration and their relation. *IEEE Trans. on Image Processing*, 1(3):322–336, 1992.

[34] D. Geman, S. Geman, C. Graffingne, and P. Dong. Boundary detection by constrained optimisation. *IEEE Trans. PAMI*, 12:609–628, 1990.

[35] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Trans. PAMI*, 6:721–741, 1984.

[36] S. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. PAMI*, 14(3):367–382, 1992.

[37] B. Gidas. A renormalization group approach to image processing problems. *IEEE Trans. PAMI.*, 11(2):164–180, 1989.

[38] G. H. Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8:155–173, 1978.

[39] S. Grossberg. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, pages 141–171, 1985.

[40] S. Grossberg and E. Mingolla. Neural dynamics of perceptual grouping: Textures, boundaries, and emergent segmentation. *Perception and Psychophysics*, pages 17–61, 1985.

[41] J. F. Haddon and J. F. Boyce. Image segmentation by unifying region and boundary information. *IEEE Trans. PAMI.*, pages 929–948, 1990.

[42] A. R. Hanson and E. M. Riseman. Processing cones: A computational structure for image analysis. In Tanimoto and A. Klinger, editors, *Structure Computer Vision.* Academic Press, New York, 1980.

[43] D. O. Hebb. *The Organization of Behaviour.* Wiely, New York, 1949.

[44] R. Held and W. Richards, editors. *Recent Progress in Perception.* Scientific American, Inc., 1976.

[45] L. Hérauld and R. Horaud. Figure-ground discrimination: A combinational optimization approach. *IEEE Trans. PAMI.*, 15:899–914, 1993.

[46] A. L. Hodgkin and A.F. Huxley. A quantitative description of membrance current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117:500–544, 1952.

[47] W. C. Hoffman. The neuron as a lie group germ and a lie product. *Quaterly Journal of Applied Mathematics*, 25:423–440, 1968.

[48] J. J. Hopfield. Neural networks and physical systems with emergent computational abilities. *Proceedings of the National Academy of Sciences USA 79*, pages 2554–2558, April 1982.

[49] J. J. Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences USA 81*, pages 3088–3092, May 1984.

[50] J. J Hopfield and D. W. Tank. Neural computation of decision in optimization problems. *Biological Cybernetics*, pages 62:141–152, 1985.

[51] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[52] T. Hsu, A. D. Calway, and R. Wilson. Texture Analysis using the Multiresolution Fourier Transform. In *Proc. 8th Scandinavian Conf. Im. Analysis*, Tromso, Norway, 1993.

[53] D. H. Hubel and D. Hunter. *Eye, Brain and Vision.* Scientific American Library, New York, 1988.

[54] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal Physiology*, pages 106–154, 1962.

[55] D. H. Hubel and T. N. Wiesel. Brain mechanisms of vision. *Scientific American*, pages 130–144, 1979.

[56] G. W. Humphreys and V. Bruce. *Visual Cognition.* Lawrence Erlbaum Associates Ltd, London, 1989.

[57] D. Hush and B. Horne. Progress in supervised neural networks: What's new since Lippmann? *IEEE Signal Processing Magazine*, pages 8–39, 1993.

[58] J. Hutchinson, C. Koch, J. Luo, and C. Mead. Computing motion using analog and binary resistive networks. *IEEE Trans. on Computer*, pages 52–63, 1988.

[59] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.

[60] A. Jain. Advances in mathematical models for image processing. *Proceedings of the IEEE*, 69(5):502–528, 1981.

[61] A. K. Jain. *Fundamental of Digital Image Processing.* New York, Berlin, 1989.

[62] E. R. Kandel and R. D. Hawkins. The biologoical basis of learning and individuality. *Scientific American*, 1992.

[63] E. R. Kandel and J. H. Schwartz. *Principles of Neural Science*. Edward Arnold Ltd, London, 1981.

[64] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. Comput. Vision*, 1:321–331, 1988.

[65] A. K. Katsaggelos. Iteartive image restoration algorithms. *Optical Engineering*, 28(7):735–748, July 1989.

[66] G. N. Khan and D. F. Gillies. Extracting contours by perceptual grouping. *Image and Vision Computing*, 10:77–88, 1992.

[67] H. Knutsson. *Filtering and Reconstruction in Image Processing*. PhD thesis, Linköping University Sweden, 1982.

[68] J. J. Koenderink. Operational significance of receptive field assemblies. *Biol. Cybern.*, 58:163–171, 1988.

[69] T. Kohonen. *Self-Organization and Associative Memory*. Springer Verlag Press, Berlin, 2 edition, 1988.

[70] A. Kulkarni. Solving ill-posed problems with artificial neural networks. *Neural Networks*, 4:477–484, 1991.

[71] M. Kunt, A. Ikonomopoulos, and M. Kocher. Second-generation image-coding techniques. *Proceedings of the IEEE*, 73(4):549–574, 1985.

[72] R. L. Lagendijk and J. Biemond. Maximum likelihood blur identification. In *SPIE, Visual Communications and Image Processing*, pages 1360–1371, Lausanne, Switzerland, 1990.

[73] R. L. Lagendijk, J. Biemond, and D. E. Boekee. Regularized iterative image restoration with ringing reduction. *IEEEPAMI*, 36:1874–1888, 1988.

[74] S. Lakshmanan and H. Derin. Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. *IEEE Trans. PAMI.*, 11:799–812, August 1989.

[75] K. J. Lang and G. E. Hinton. A time-delay neural network architecture for speech recognition. Technical Report CMU-CS-88-152, Carnegie-Mellon University, 1988.

[76] A. Lapedes and R. Farber. Nonlinear signal processing using neural networks: Prediction and signal modeling. Technical Report LA-UR-87-2662, Los Alamos National Laboratories, Los Alamos, New Maxico, 1987.

[77] D. S. Levine. *Introduction to Neural and Cognitive Modeling*. Lawrence Erlbaum Assoc., 1991.

[78] J. S. Lim. *Two-Dimensional Signal and Image Processing*. Prentice Hall, 1990.

[79] T. P. Linderberg. Scale-space for discrete signals. *IEEE Trans. PAMI.*, pages 234–254, 1990.

[80] R. Linsker. Self-organization in a perceptual network. *Computer*, pages 105–117, March 1988.

[81] R. P. Lippman. An introduction to computing with neural nets. *IEEE ASSP Magzine*, pages 4–22, April 1987.

[82] Y. Lu and R. C. Jain. Reasoning about edges in scale space. *IEEE Trans. PAMI.*, 14(4):450–468, April 1993.

[83] M. R. Luettgen, W. C. Karl, and A. Willsky. Efficient multiscale regularization with applications to the computation of optical flow. *IEEE Trans. on Image Processing*, 3(1):41–63, January 1993.

[84] J. Maeda and T. Sakabe. Estimation of Optimal Regularization Parameter in an Iterative Anisotropic Image Restoration method. In *Proc. 8th Scandinavian Conf. Im. Analysis*, pages 191–198, Tromso, Norway, 1993.

[85] M. Mahowald and C. Mead. The silicon retina. *Scientific American*, pages 40–46, 1991.

[86] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. PAMI.*, 11(7):674–693, July 1989.

[87] B. S. Manjunath and R. Chellappa. A unified approach to boundary perception: Edges, textures and illusory countours. *IEEE Trans. on Neural Networks*, pages 96–108, January 1993.

[88] D. Marr. *Vision*. Freeman Press, San Fransisco, C.A., 1982.

[89] A. Martelli. An application of heuristic search methods to edge and contour detection. *Commun. Assoc. Comput. Mach.*, 19:73–83, 1976.

[90] J. L. McClelland, D. E. Rumelhart, and the PDP Research Group. *Parallel Distributed Processsing, Volumn 2:Psychological and Biological Models*. MIT Press, Cambridge, Massachusetts, 1986.

[91] W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 1943.

[92] R. J. McEliece, E. C. Posner, E. R. Rodemich, and S. S. Venkatesh. The capacity of the Hopfield associative memory. *IEEE Trans. on Information Theory*, IT-33, 1986.

[93] E. Mjolsness, C. D. Garrett, and W. L. Miranker. Multiscale optimization in neural nets. *IEEE Tran. On Neural Networks*, 2(2):263–274, March 1991.

[94] P. Montesinos and J. P. Fabre. Perceptual grouping through recursive optimization. In *Proc. 8th Scandinavian Conf. Im. Analysis*, Tromso, Norway, 1993.

[95] K. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Trans. on Neural Networks*, 1:4–27, 1990.

[96] C. R. Noback and R. J. Demarest. *The Human Nervous System Basic Principles of Neurobiology.* McGraw-Hill, New York, 1981.

[97] J. K. Paik and A. K. Katsaggelos. Image restoration using a modified Hopfield network. *IEEE Trans. on Image Processing*, 1(1):49–63, January 1992.

[98] F. Palmieri. A backpropagation algorithm for multilayer hybrid order statistic filters. In *Proc. ICASSSP*, Glasgow, Scotland, 1989.

[99] Y. Pao. *Adaptive Pattern Recognition and neural Networks.* Addison Wesley Publishing Company, Inc., New York, 1989.

[100] A. Papoulis. *Signal Analysis.* McGraw-Hill, New York, 1977.

[101] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, New York, 1991.

[102] P. Parent and S. W. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Trans. PAMI.*, 11(8):823–839, August 1989.

[103] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. PAMI.*, 12(7):629–639, July 1990.

[104] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317 26:314–319, Sept 1985.

[105] A. Psarrou and H. Buxton. A neural network approach to the computation of vision algorithms. *First IEE International Conference on Artificial Neural networks*, pages 67–70, October 1989.

[106] W. Qian, M. Kallergi, and L. P. Clarke. Order statistic-neural network hybrid filters for gamma camera-bremsstrahlung image restoration. *IEEE Trans. on Medical Imaging*, pages 58–64, March 1993.

[107] S. Rajala and R. deFigueiredo. Adaptive nonlinear image restoration by a modified kalman filtering approach. *IEEE Trans. Acous. Speech Sig. Proc.*, ASSP-29:1033–1042, 1981.

[108] S. J. Reeves and R. Mersereau. Optimal estimation of the regularization paramter and stabilizing functional for regularized image restoration. *Optical Engineering*, 29(5):446–454, 1990.

[109] A. Rosenfeld and A. C. Kak. *Digital Picture Processing.* Academic Press, New York, 2 edition, 1982.

[110] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Trans. Comput*, c-20:562–569, 1971.

[111] D. E. Rumelhart and J. L. McClelland. *Parallel Distributed Processsing, Volumn 1:Foundations.* MIT Press, Cambridge, Massachusetts, 1986.

[112] T. D. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2:459–473, 1989.

[113] E. Saund. *The Role of Knowledge in Visual Shape Representation.* PhD thesis, MIT Artificial Intelligence Laboratory, 1988.

[114] T. Sejnowski and G. Hinton. Separating figure from ground with a Boltzmann machine. In M. Arbib and A. Hanson, editors, *Vision, Brain and Cooperative Computation.* MIT press, Cambridge, MA, 1985.

[115] T. Sejnowski and C. Rosenberg. NETtalk : A parallel network that learns to read aloud. Technical Report JHU/EECS-86/01, The Johns Hopkins University Electrical Engineering and Computer Science, 1986.

[116] A. Shaáshua and S. Ullman. Structural saliency: The detection of globally salient structures using a locally connected network. In *Proc. IEEE International Conf. Computer Vision,* Tampa, FL, 1988.

[117] C. J. Shatz. The developing brain. *Scientific American,* pages 35–41, 1992.

[118] T. Simchony, R. Chellappa, and Z. Lichtenstein. Pyramid implementation of optimal-step conjugate-search algorithms for some low-level vision problems. *IEEE Trans. Sys, Man Cyber.,* pages 1408–1425, 1989.

[119] H. Tan, S. B. Gelfand, and E. J. Delp. A cost minimization approch to edge detection using simulated annealing. *IEEE Trans. PAMI.,* 14:3–18, January 1992.

[120] D. Terzopoulos. Regularsation of inverse visual problems involving discontinuities. *IEEE Trans. on PAMI,* PAMI-8, No 4:413–424, July 1986.

[121] M. P. Todd. *Image Data Compression Based On a Multiresolution Signal Model.* PhD thesis, The University of Warwick UK, 1989.

[122] D. L. Toulson and J. F. Boyce. Segmentation of MR images using neural nets. *Image and Vision Computing,* 10:324–328, 1992.

[123] H. J. Trussell and M. R. Civanlar. The feasible solutions in signal restoration. *IEEE Trans. on Acoust, Speech, Signal processing,* ASSP-32, No 4:201–212, 1984.

[124] H. J. Trussell and B. R. Hunt. Sectioned methods for image restoration. *IEEE Trans. Acous. Speech Sig. Proc.,* ASSP-26, No 2:157–164, 1978.

[125] S. Urago, J. Zerubia, and M. Berthod. A Markovian model for contour grouping. Technical Report N2122, Institut National De Recherche En Informatique Et Automatique, France, 1994.

[126] F. S. Werblin. The control of sensitivity in the retina. *Scientific American,* 228:70–79, 1973.

[127] R. Wilson and A. H. Bhalerao. Kernel designs for efficient multiresolution edge detection and orientation estimation. *IEEE PAMI,* 14(3):384–390, 1992.

[128] R. Wilson, S. Clippingdale, and A.H. Bhalerao. Robust estimation of local orientations in images using a multiresolution approach. In *SPIE, Visual Communications and Image Processing,* pages 1393–1403, Lausanne, Switzerland, 1990.

[129] R. Wilson and G. H. Granlund. The uncertainty principle in image processing. *IEEE Trans. on PAMI*, 6:758-767, 1984.

[130] R. Wilson and G. H. Granlund. Uncertainty and inference in the visual system. *IEEE Trans. PAMI.*, 18:305-312, 1988.

[131] R. Wilson and M. Spann. *Image Segmentation and Uncertainty.* Pattern Recognition and Image Processing Series. Research Studies Press Ltd, 1988.

[132] A. P. Witkin. Scale-space filtering. In *Proc. IJCAI*, Karlsruhe, 1983.

[133] J. W. Woods. Two-dimensional discrete Markovian fields. *IEEE Trans. on Information Theory*, IT-18:8-16, March 1972.

[134] Z. Wu. Optical flow estimation by using the artificial neural network under multi-layers. In *First IEE International Conference on Artificial Neural networks*, pages 76-80, October 1989.

[135] H-C. Yang and R. Wilson. Orientation-directed edge detection using a Hopfield neural network. In *Proc. 8th IEEE workshop on Image and Multidimensional Signal processing*, Cannes, France, 1993.

[136] S. J. Yeh, H. Stark, and M. I. Sezan. Hopfield-type neural networks. In A. K. Katsaggelos, editor, *Digital Image Restoration*, volume 23, chapter 3. Springer-Verlag, Germany, 1991.

[137] S. Yu and W. Tsai. Relaxation by the Hopfield neural network. *Pattern Recognition*, 25(2):197-209, 1992.

[138] S. Zeki. The visual image in mind and brain. *Scientific American*, pages 43-50, 1992.

[139] J. Zerubia and R. Chellappa. Mean field approximation using compound Gauss-Markov random field for edge detection and image restoration. *Proceedings ICASSP 90*, pages 2193-2196, 1990.

[140] J. Zerubia and R. Chellappa. Mean field annealing using compound Gauss-Markov random fields for edge detection and image restoration. *IEEE Trans. on Neural Networks*, 8(4):703-709, 1993.

[141] Y-T Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins. Image restoration using a neural network. *IEEE Tran. ASSP*, 36(7), July 1988.

[142] S. Zucker, R. Hummel, and A. Rosenfeld. An application of relaxation labeling to line and curve enhancement. *IEEE Trans. Computer*, C-26(7), July 1977.

[143] S. W. Zucker. A biologically motivated approach to early visual computations: Orientation selection, texture, and optical flow. In J. Kittler, editor, *Proc. 4th BPRA conf. on PR, lecture notes in CS*, volume 301, pages 418-428, March 1988.